
ATARI ST
COMPUTER
Die Fachzeitschrift für den ATARI-ST Anwender.
Juni 90
DM 8,-
Ös. 64,-
Sfr. 8,-
6
Resource Construction Sets
Die Bausteine des ST
Tempus Word
Ein Vorabbericht
Manhatten ST
Der Super-Rechner
Easybase
Datenbank light
Spieledesign auf dem ST
ST Report
Luftraumnutzungszentrale Frankfurt
Große Leserumfrage
Preise im Wert von DM 15000,- zu gewinnen
ATARI

EDITORIAL

Commodore greift an

Was ATARI seit vielen Monaten nicht geschafft hat, wird Commodore im Juni bewältigen: Der AMIGA 3000 wird auf den Markt geworfen. Dieses Gerät wird eine ernstzunehmende Konkurrenz zum TT werden - sofern dieser denn endlich bei einem oder zwei von ATARI auserwählten Fachhändlern erscheint. Sollte der TT nicht innerhalb weniger Wochen ausgeliefert werden, dürfte ATARI es schwer haben, das Gerät im Markt zu etablieren. Der AMIGA 3000 wartet mit beeindruckenden Daten auf: 68030-Prozessor mit 16 oder 25 MHz Systemtakt, Coprozessor 68881 oder 68882, 40 bis 100 MB-SCSI-Platte, herausgeführter SCSI-Bus, eine NeXT-ähnliche Oberfläche (Workbench 2.0), Stereo-Sound, Video-Slot, 8 Erweiterungsports für Steckkarten, 2 MB RAM serienmäßig, erweiterbar auf bis zu 18 MB. Das Gerät arbeitet sowohl intern als auch extern mit 32 Bit. Anschlüsse für einen MultiSync sind vorhanden - die Bildschirmauflösung geht bis zu 1448x566 Pixeln mit 4 Farben bzw. Graustufen. Die kleine Ausführung wird zwischen 5000,- und 6000,- DM kosten...

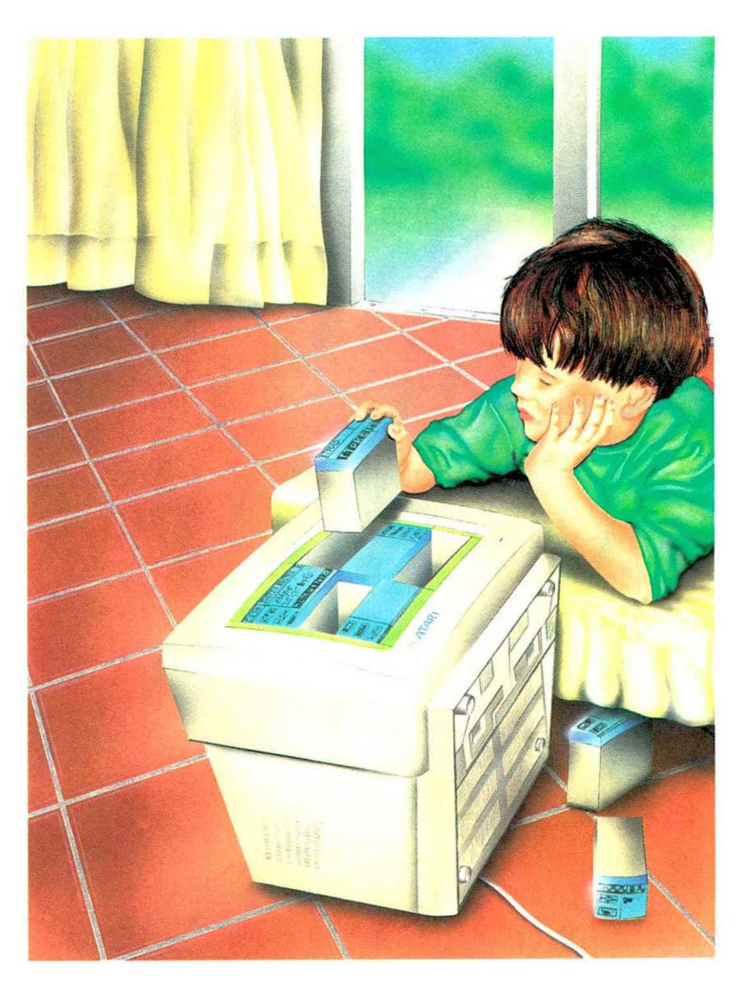
Auch wenn viele ATARI-Freaks mir bei diesem Satz die Freundschaft kündigen werden, sehen Sie bereits an den oben aufgeführten Daten, daß der TT gegen den A3000 manchmal schon etwas schmächtig aussieht. Wenn ATARI den TT noch verkaufen will, sollte er schnell auf den Markt kommen - sehr schnell. Vielleicht sollte man im Hause ATARI bei der Preisgestaltung auch noch etwas umdenken?

Martin Pittelkow

INHALT

SOFTWARE

GRUNDLAGEN



Konstruktive Resourcen - Vergleich von RCS'

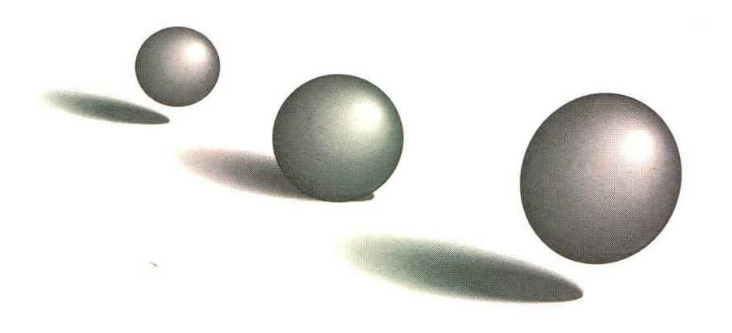
Eines der wichtigsten Utilities beim Erstellen von GEM-Programmen ist ohne Zweifel das Resource Construction Set. Dieses schmucke Programm hat inzwischen ein paar Geschwister und einen Nachfolger bekommen, die vorgeben, es mit ihm aufnehmen zu können oder besser zu sein. Dies ist sicherlich ein Grund, diese unterschiedlichen Resource Construction Sets einmal unter die Lupe zu nehmen.

Seite 20



Vor einiger Zeit beglückte CCD seine Kunden mit einer verheißungsvollen Botschaft. Tempus Word, sei nun verfügbar. Zwar eine 0.9-Version, in der noch einiges fehle. Aber immerhin: Der Preis betrage nur 450 DM, die Version 1.0 würde 200 Märker mehr kosten. Entscheiden sollte man sich zügig, denn nur 500 0.9-Exemplare sollten ausgegeben werden. Wir habe die ehrenvolle Aufgabe, Ihnen dieses Programm näherzubringen.

Seite 38



Professionelles Spieledesign auf dem ST

Kennen Sie BOLO oder esprit? Der Autor dieser beiden erfolgreichsten Spieleprogramme auf dem ATARI ST läßt uns erstmals und exklusiv in ST-Computer hinter die Kulissen einer professionellen Spielentwicklung schauen.

Seite 132



Manhatten ST - Hat man 'en ST?

Wenn man den "Manhatten ST" vor sich stehen sieht, glaubt man nicht mehr, daß man einen ST hat. Manhatten ST ist ein High-End-Tower-ST, gegen den ein Mega ST wie ein ZX 81 wirkt. Viel interessanter ist aber, was im Tower seinen Dienst verrichtet...

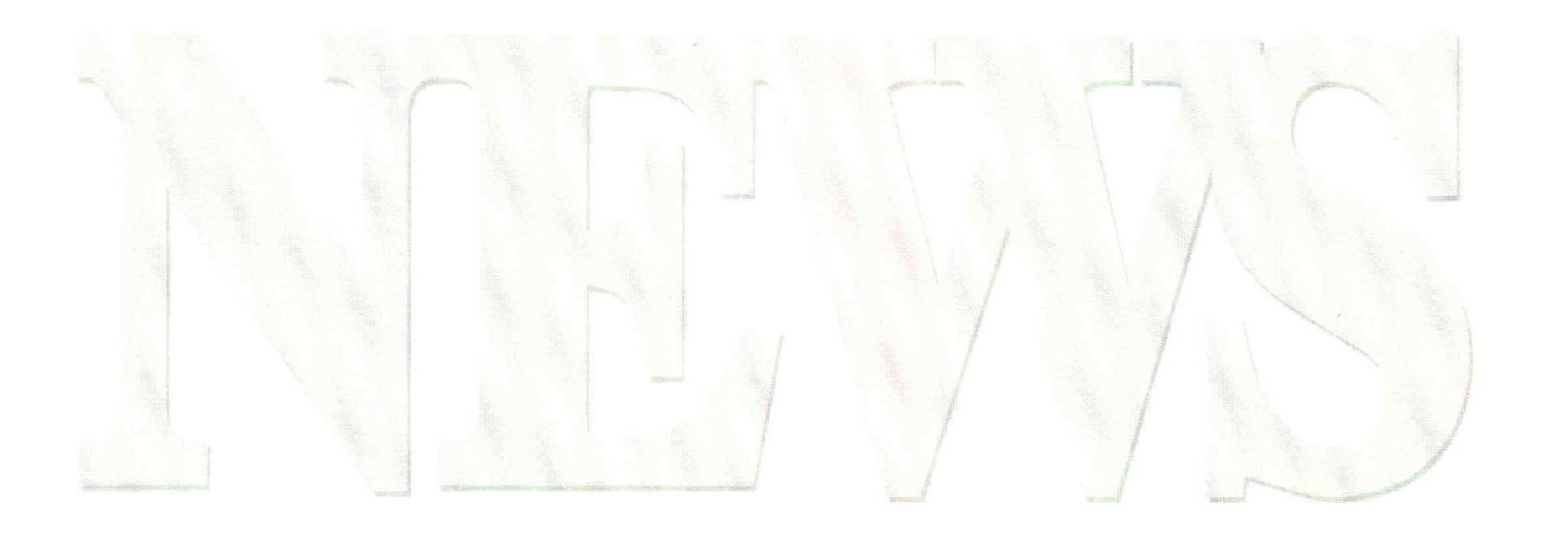
Seite 132

PROGRAMMIERPRAXIS

HARDWARE

AKTUELLES

RUBRIKEN



Modula-2 HM plus 3.1

Hänisch Modula ist ab sofort in der Version HM plus 3.1 verfügbar. HM plus wird in zwei Ausführungen angeboten: Juniorversion (DM 298,-) und Profiversion (DM 770,-). Die Ausführungen unterscheiden sich hinsichtlich Lieferumfang und Lizenzbestimmungen.

Schwab Software
Obermarkstraße 72
4600 Dortmund 30
Tel. (0231) 486460

technobox Drafter

Nach den neuesten Erkenntnissen der Benutzerführung und Software-Entwicklung hat die Technobox Software GmbH ein Zeichenprogramm auf den Markt gebracht, das aus dem Vorgängerprogramm CAMPUS CAD und dem späteren Technobox CAD/1 entstand. Der Drafter bietet als erstes 2D-Zeichensystem die Möglichkeit der perspektivischen Konstruktion mit einer entsprechenden Genauigkeit. Die vollständige Bemaßung ist Standard im Programm. Der Drafter unterstützt schon jetzt eine Farbauflösung mit 16 Farben bei 640x400 Pixeln. Für die Einbindung in technische Dokumentationen kann die Ausgabe jeder Zeichnungsdatei im Metafile-Format erfolgen. Völlig neu ist, daß man eine Zeichnung auch ausschnittsweise und unter Berücksichtigung der Strichstärken in das DTP-Programm übernehmen kann. Somit ist der Drafter das ideale Programm, wenn es darum geht, technische Zeichenungen mit einer sehr guten Qualität in Texte oder Dokumentationen zu integrieren.

Technobox Software GmbH
Kornharpener Straße 122a
4630 Bochum 1
Tel. (0234) 503060

Calamus-Fonts

Für Calamus werden nun von der Firma W. Wohlfahrtstätter und J. Ohst EDV GbR Vektor-Fonts angeboten. Alle Zeichensätze lassen sich problemlos belichten, da sie durch eine eigene Seriennummer gesichert werden. Die Calamus-Seriennummer ist somit nicht erforderlich. Die Zeichensätze sind ab DM 39,- erhältlich.

W. Wohlfahrtstätter und J. Ohst
EDV GbR
Nelkenstraße 2
4053 Jüchen 2
Tel. (02164) 7898

Platon V1.4

(Bericht in in ST 03/90)

Leider waren im Testbericht über Platon in der Ausgabe 03/90 einige Unklarheiten, die wir hiermit verbessern wollen:

* Auch beim Druckertreiber (DRUCKER.PRG) ist die Möglichkeit gegeben, die Ausgabegröße stufenlos einzustellen.

* Da der NEC-Drucker kompatibel zum LQ-1000 ist, existiert selbstverständlich auch ein NEC-Treiber. Sogar die hohe Auflösung von 360x360 Punkten wird vom Treiber unterstützt.

* Inzwischen wird Platon mit Druckertreibern für den ATARI-Laser-Drucker und den HP-DeskJet ausgeliefert.

* Die angesprochenen Probleme mit TOS 1.4 wurden beseitigt.

VHF-Computer
Meurener Weg 115a
7030 Böblingen
Tel. (07031) 289211

INTERLINK ist eine Standleitung in die UDSSR

Die GTC TeleCommunication in Stuttgart bietet ab sofort als erstes und einziges Unternehmen in Europa eine Standleitung nach Moskau an, die weltweit von allen Teilnehmern des GTC-Netzes genutzt werden kann. Der Name des sowjetischen Joint Ventures und der GTC TeleCommunication: 'INTERLINK USSR'. Damit können westliche Firmen und Institutionen ab sofort problemlos mit ihren Geschäftspartnern und Niederlassungen in der gesamten Sowjetunion kommunizieren. Die Standleitung der GTC teleCommunication beseitigt das bisher große Problem der ungenügenden oder gar fehlenden Kommunikationsverbindungen in die UDSSR. Das bisherige Kommunikationsproblem mit dem Osten, stundenlanges, oftmals vergebliches Wählen über Telefon oder Fax, ist durch das bisher in Europa einzigartige System der 'Computer Aided Communication' (CAC) denkbar einfach geworden. Alles, was benötigt wird, ist ein Computer wie der ST und eine Telefonverbindung via Modem. Die monatliche Grundgebühr beträgt DM 40,-, jede Online-Minute DM 0,25. Gebühren für Netzdienste wie Telexversand oder Datentransfer richten sich nach Art und Umfang.

GTC TeleCommunication
- Generalrepräsentanz -
Oberau 77
7800 Freiburg/Breisgau
Tel. (0761) 381051

Neuer BEST-Distributor für Deutschland

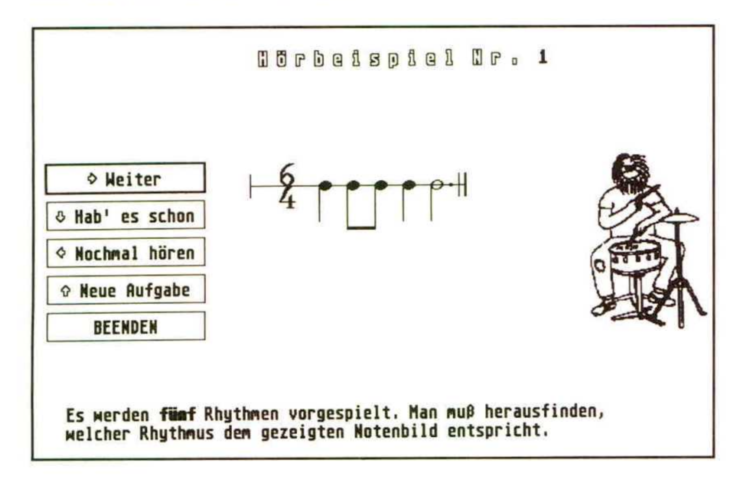
Die neue Distribution für BEST-Modems in Deutschland hat die Firma Telekommunikation Kaben Riis (TKR) in Kiel übernommen, sie übernimmt damit die Aufgabe der Firma Bieling Computersysteme. TKR vertreibt das gesamte Angebot von BEST wie beispielsweise die Modems 2400 L (300, 1200, 2400 Baud), 2400 EC (MNP 5) und viele mehr. Aber auch das BTX-Programm Multiterm pro kann über TKR bezogen werden.

TKR
Projensdorfer Straße 14
2300 Kiel
Tel. (0431) 337881

Hörtrainig mit dem "Computerkolleg Musik"

Unter dem Titel "Computerkolleg Musik" ist beim Musikverlag B. Schotts Söhne für alle ATARI ST ein Programmpaket mit vier Kursen zur gehörbildung im Selbststudium erschienen. Die von der Osnabrücker CAMI-Group entwickelten Programme Intervalle, Skalen, Rhythmen und Akkorde arbeiten adaptiv, d.h. der Computer paßt sich automatisch dem Leistungsstand des Übenden an. Alle Antworten werden vom Computer kontrolliert, kommentiert und differenziert ausgewertet. Die Kurse gliedern sich in einen Info-Teil sowie in Vor- und Hauptübungen und beinhalten spezielle MIDI-Übungen am Instrument, die eine musikpraktische Umsetzung des gelernten Übungsstoffes gestatten. Auf Wunsch läßt sich der erzielte Leistungsstand in einem persönlichen Zertifikat dokumentieren und ausdrukken. Das komplette Programmpaket mit den vier Grundkursen ist bis zum August zum Subskriptionspreis von DM 298,- (danach DM 398,-) erhältlich.

imaC-Systemlösungen
Gröbelweg 13
4500 Osnabrück
Tel. (0541) 597419



Dipsi jetzt in Version 1.10

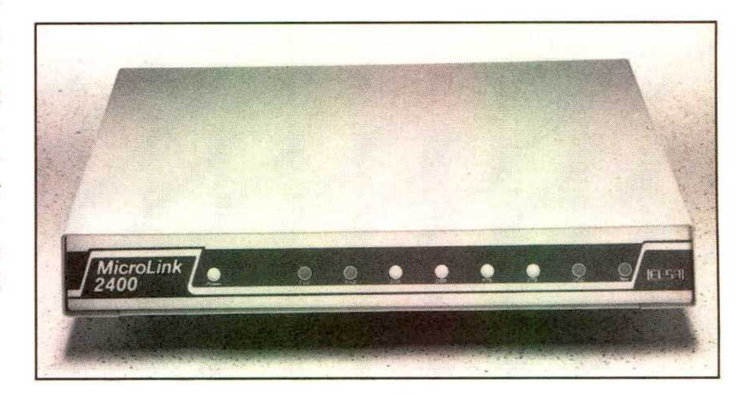
Dipsi gibt es jetzt in der Version 1.10. Die Version hat eine neue Funktion zur Ausgabe bekommen: "FREI" hat die Möglichkeit, jedes Datenfeld eines Datensatzes frei zu positionieren. Damit ist die Möglichkeit gegeben, einen Vordruck oder ein Karteiblatt von Dipsi mit Daten ausfüllen zu lassen. Der Text kann sowohl vertikal als auch horizontal gedruckt werden. Weiterhin ist jetzt eine Demoversion mit Kurzbeschreibung für DM 20,- erhältlich.

ERSE Software
Hauptstraße 59
2278 Wittdün / Amrum
Tel. (04682) 2939

ELSA-Modem in der DDR zugelassen

MicroLink 2400T2, in der Bundesrepublik eines der erfolgreichsten zugelassenen 2400 Bit/s-Modems, wurde jetzt vom Institut für Post- und Fernmeldewesen der DDR zum Einsatz im Telefonwählnetz zugelassen. Wegen der bekannten Schwächen des dortigen Telefonnetzes wurde die Zulassung unter der Auflage erteilt, daß bei Übertragungsgeschwindigkeiten oberhalb von 300 bit/s der Gebrauch des Fehlerkorrekturverfahrens MNP (Microcom Networking Protocol) vorgeschrieben ist. Bei der Zulassung, die vom ELSA-Vertriebspartner Motorola (Darmstadt) betrieben wurde, handelt es sich um die zweite Allgemeinzulassung in der DDR für ein V.22bis-Modem (nach einem ungarischen Modell) und um die erste Zulassung für einen westdeutschen Modemhersteller überhaupt. Das Modem erhielt die Zulassungsnummer 32390474. ELSA wird in der DDR von der Firma HELTEC (Potsdam) vertreten.

ELSA GmbH
Monheimsallee 53
5100 Aachen
Tel. (0241) 477890



Jugend und Computer

Einen großen Grafikwettbewerb für die etwa eine Million Computerfreaks in Nordrhein-Westfalen startet die LBS. Beteiligen können sich alle, die zwischen 13 und 25 Jahren jung sind, in NRW wohnen und mit Homecomputer oder PC - egal welchen Systems - arbeiten. Mitmachen können aber auch Computerclubs oder Schulklassen mit einer "Gemeinschaftsgrafik". Für die 110 besten Arbeiten sind interessante Preise rund um den Computer ausgesetzt (erster bis dritter Preis je eine Woche Urlaub in einem attraktiven Computercamp). Ermittelt werden die Preisträger von einer prominenten Jury, darunter der bekannte Düsseldorfer Computerkünstler Jürgen LIT Fischer und Prof. Henning Freiberg, einer der führenden Kunstpädagogen für Computer-Anwendungen. Aufgabe der Wettbewerbsteilnehmer ist es, den Begriff "Vario" (Motto: "Vario= Variationen= Varianten") frei in eine Computer-Grafik unzusetzen. Dabei ist es möglich, das Wort "Vario" in verschiedenen Schriften und Farben zu gestalten oder mit Hilfe des Computers in einer Animation zu zeigen. Die genauen Teilnahmebedingungen können seit dem 8. Mai bei allen LBS-Beratungsstellen in Nordrhein-Westfalen abgeholt oder bei der LBS angefordert werden. Einsendeschluß ist der 1. August 1990. Die Computer-Grafiken sollen auf Diskette gespeichert werden. Zusätzlich sind Angaben zu verwendetem Computer, zusätzlicher Peripherie (z.B. Scanner, Grafiktablett etc.) und benutztem Programm erforderlich.

LBS
Himmelreichallee 40
4400 Münster
Tel. (0251) 41202

Computerclub Magdeburg / Sektion ATARI

Für alle ATARI-Freaks auf der DDR-Seite des Lands sei der Computerclub Magdeburg / Sektion ATARI erwähnt, der sich momentan aus einer Gruppe von mehr als 150 begeisterten Mitgliedern zusammensetzt, die sich alle 14 Tage treffen. Es wurde bereits ein Soundsampler gebaut, der auf einem Treffen in Verbindung mit einem Walkman und einem Mikrofon vorgeführt wurde. Eine MIDI-Schnittstelle für den ST wurde gebaut und im Club in Verbindung mit einem Yamaha-Soundstudio erprobt. Der Bau von EPROM-Steckmodulen, auf denen Programme, Programmiersprachen, DOS und Laderoutinen gespeichert sind, ist sehr gut angelaufen. Mehrere Mitglieder rüsteten ihren Computer speicherseitig auf, so daß umfangreiche RAM-Disks Anwendung finden konnten. Auch Vorträge kommen auf den Treffen nicht zu kurz und erweitern das Wissen der Mitglieder. Am 9. Juni 1990 wird das erste deutsch-deutsche Computertreffen durchgeführt. Weitere Infos bei

Computerclub an der Technischen Universität Magdeburg
Sektion ATARI / Herrn Bähr
B. Hahn
Postlagernd HPA
3010 Magdeburg

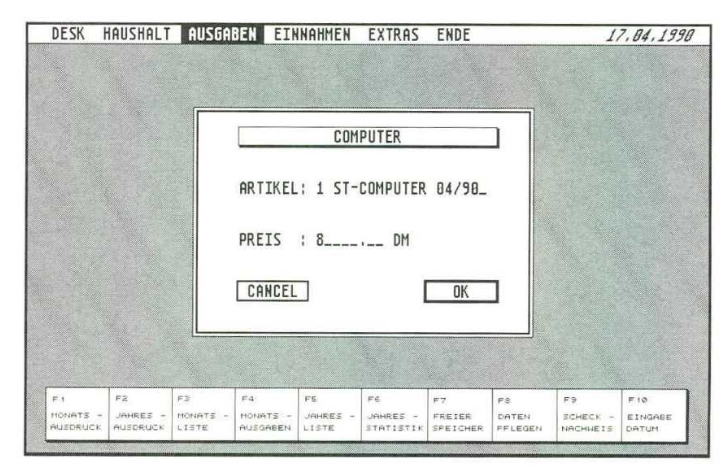


SW-Haushalt und SW-Video

Die Firma SW-Software bietet ab sofort zwei neue Programme für den privaten Bereich an. Bei dem ersten Programm SW-Haushalt handelt es sich um eine private Haushaltsbuchführung, die jedem Computerbesitzer auch ohne Buchführungskenntnisse die Möglichkeit bietet, alle anfallenden Einnahmen und Ausgaben eines Monats bzw. eines gesamten Jahres voll menügesteuert zu verwalten. Das Programm kostet DM 69,- inklusive ausführlicher Bedienungsanleitung.

Das zweite Programm, SW-Video, ist ein Programm zur Verwaltung von bis zu 2500 Videocassetten nach Filmtitel, Cassettennummer, Filmgenre usw. Natürlich können auch alle Daten auf dem Drucker ausgegeben werden. Ebenfalls besteht die Möglichkeit, alle Etiketten zu bedrucken. SW-Video ist ebenfalls sehr einfach durch Menüsteuerung zu bedienen und kostet inklusive Handbuch DM 59,-.

SW-Software
Beethovenstraße 10
7938 Oberdischingen
Tel. (07305) 8325



Label Expert V5.8

Ab sofort ist die neue Version 5.8 des Etikettendrucksystems Label Expert verfügbar. Neben den 18 Standardetiketten können jetzt mit einem Malprogramm eigene, neue Etiketten gestaltet und fest in Label Expert eingebaut werden. Die Eingabemasken wurden außerdem erweitert. Einige Funktionen können jetzt auch über Icons und Funktionstasten erreicht werden. Im Lieferumfang enthalten ist die Diskettenverwaltung Disk Master sowie das PD-Programm Public Painter. Das Programm kostet incl. Anleitung DM 39,-, die Update-Kosten für alle registrierten Benutzer betragen DM 10,-.

Jörg Trojan
Amselweg 9
5216 Niederkassel 3
Tel. (02208) 5364

OnChip-Systems

Der Spezialist für analog-digitales Chipdesign wird ab sofort im deutschsprachigen Raum vertreten durch MICON AUDIO electronics GmbH. Die in der Musikelektronik bekannte CEM 33xx-Reihe (VCA, VCF, VCO, Signalprozessor) und die CEM 55xx-Reihe (8fach S&H + Mux, Signalprozessor) sind bereits auch in anderen Industriebereichen etabliert. Weitere Produkte sind: SAM 8905 (digitaler Audio-Signalprozessor), PT212AT (Vollduplex-Modem), PCT13/13E (drahtloses Telefon Basis-System) und TCT14 (drahtloses Telefon Handset-System).

MICON AUDIO electronics GmbH
Bremer Straße 11
6236 Eschborn
Tel. (06196) 46923

ATARI-Road-Show

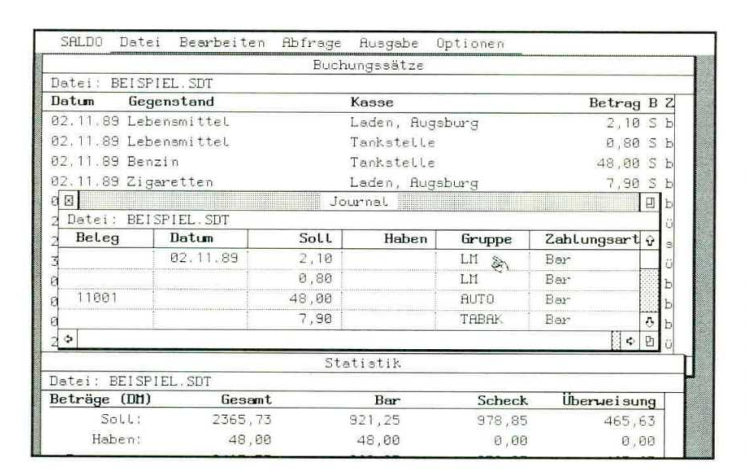
In den vier Städten Hamburg, Düsseldorf, Frankfurt und München wurden unter der Schirmherrschaft von ATARI und unter der Leitung von Erich Grikscheit mit den Firmen GMA, 3K, DMC, Richter Distributor, ADI, CCD, TmS, Compo, Shift, Application Systems und WordPerfect DTP- und Händlertreffen veranstaltet. Themen dieser Veranstaltung waren: Schneideplotten mit dem ST, Bildverarbeitung und Digitalisierung, Schriftenerkennung für die Texterfassung, Textverarbeitung für DTP und Scanner. An Software wurde vorgeführt: That's Write, Adimens, WordPerfect, Script, Arabesque, Calamus, Retouche, Cranach, GMA Plot, Sherlook in Version 3.0. Nach einhelliger Meinung der an der Roadshow beteiligten Firmen und Besucher war diese Veranstaltung ein voller Erfolg. Händler konnten neue und gründliche Informationen für ihre Kunden mitnehmen, so daß auch nächstes Jahr dieses Treffen stattfinden wird. Die Termine werden rechtzeitig bekanntgegeben. Die teilnehmenden Firmen danken noch einmal der Firma ATARI für die gelungene und kooperative Veranstaltung im Sinne und Nutzen für die professionellen Endkunden.

Info: Richter Distributor / Erich Grikscheit

Saldo

Das Problem des Geldverkehrs ist nicht nur eine Domäne großer Firmen und gerissener Manager - die kleine, aber aufwendige Kontrolle der Finanzen in unserer Welt ist schon oft als eine Nowendigkeit erwiesen worden. Saldo ist ein Programm, das Ihnen erlaubt, auf bequeme Weise die Kontrolle über Ihr schwer verdientes Geld zu bekommen. Man kann Saldo für rein private Zwecke zur ständigen Kontrolle der eigenen finanziellen Lage verwenden, es aber ebenso auch als Einnahme- und Überschußverwaltung für eine gewerbliche Tätigkeit einsetzen. Durch eine eingebaute Paßwortfunktion kann man die privaten Daten vor fremdem Zugriff schützen. Weiterhin bietet Saldo zahlreiche Möglichkeiten der Manipulation eingetragener Daten. Sie können sortiert, gesplittet oder auf dem Drucker ausgegeben werden usw. Durch ein komfortables Abfragesystem kann der Datenbestand nach verschiedenen Kriterien sortiert werden. So kann man zum Beispiel die Einzelausgaben für Autozubehör, die man in den letzten drei Wochen per Scheck bezahlt hat, betrachten, oder den Kinokonsum eines bestimmten Zeitraums kontrollieren, oder wann man geborgtes Geld zurückzahlen möchte. Eine praktische Anwendung im Falle einer Einnahme- und Überschußverwaltung ist die Aufschlüsselung der Daten nach Gegenständen, Konten und Steuerkennziffern. So läßt sich bequem eine Umsatzsteuervoranmeldung anfertigen. Saldo bietet jedem eine Vielzahl von Möglichkeiten, die Finanzen zu kontrollieren; sei es für den Haushalt oder ein eingenes Gewerbe. Saldo kostet DM 79,-.

BELA Computer Layout- und Vertriebs GmbH
Unterortstraße 23-25
6236 Eschborn
Tel. (06196) 481944



Erholung für den Laser-Drucker

Mit dem Laserinterface der Firma Digital Image ist es DTP-Anwendern nun möglich, auch während des DMA-Betriebs den ATARI SLM 804-Laser-Drucker ausgeschaltet zu lassen. Die Betroffenen kennen das Problem: Bei inaktiviertem Drucker ließ sich bislang das Programm nicht booten. Das ungenutzt angeschaltete Gerät vergeudete Strom und stellte sicher für manchen eine Lärmbelästigung (wegen des Gebläses) dar. Durch Integration des Laserinterfaces - es wird einfach zwischen den DMA-Treiber und den Drucker gesteckt - kann Abhilfe geschaffen werden. Der Laser-Drucker kann zunächst ausgeschaltet bleiben und später dann bei Bedarf jederzeit aktiviert werden. Eine versehentliche Erteilung eines Druckbefehls trotz ausgeschalteten Druckers wird von professionellen DTP-Programmen mit einer Fehlermeldung beantwortet, nach Einschalten des Druckers und einer kurzen Aufheizzeit kann der Befehl problemlos wiederholt werden.

Digital Image
Postfach 1206
6096 Raunheim
Tel. (06142) 22636

"Bessere Desktop-Alternative" jetzt NOCH besser

Neodesk ist ab sofort umweltfreundlich! Mit dem rettenden Papierkorb können Sie gelöschte Dateien jederzeit wiederbeleben. Der neue Papierkorb ist die erste Anwendung der Kommunikationsschnittstelle von Neodesk 2.05. Sie erlaubt es anderen Programmen, auf die Funktionen von Neodesk zurückzugreifen. Das hat mehrere Vorteile: Neue Funktionen für Neodesk können leichter entwickelt werden, weil die Programme auf die Bibliothek von Neodesk zurückgreifen können. Die Erweiterungen belegen weniger Speicherplatz, weil Routinen, die in Neodesk vorhanden sind, nicht ein zweites Mal Platz im Speicher verbrauchen. Die neuen Funktionen können vollständig integriert werden. Die Symbole von Erweiterungen können auf das Desktop gelegt und die Programme darüber angesprochen werden. So ist es auch möglich, Dateien auf das Symbol zu legen und sie so an das neue Modul weiterzugeben. In Kürze wird auch ein Entwicklerpaket angeboten, das nicht nur eine genaue Beschreibung der Schnittstelle beinhaltet, sondern auch weitere Beispiele. Der rettende Papierkorb ist gegen Einsendung von DM 10,- für die Diskette und den Versand erhältlich.

Computerware
Weißer Straße 76
5000 Köln 60
Tel. (0221) 392583

Applikant ST

Das in Ausgabe 4/90 vorgestellte Programm "Applikant ST" ist leider mit einer falschen Adresse veröffentlicht worden. Die neue Adresse lautet:

MCP-Microcom-Promotions-TM-Software Tauchnitz&Stahl
Bodo-Norbert Tauchnitz & Günter Stahl
Kehrwieder 4
4100 Duisburg 26

ATARI-Messe 1990

Auch in diesem jahr ist es bald wieder soweit: Dieses Jahr findet zum vierten Mal die ATARI-Messe statt. Man kann ohne zu übertreiben behaupten, daß diese Messe das Mekka aller ATARI-Computer-Anwender ist. In diesem Jahr werden ungefähr 200 Aussteller das Neueste vom Neuen der Hard- und Software-Palette rund um den ATARI-Computer anbieten. Nach der überaus erfolgreichen ATARI-Messe 1989 (155 Aussteller und 35000 Besucher) muß dieses Jahr das Ausstellungsgelände erweitert werden. Aufgrund der großen Nachfrage wurde das Gesamtangebot auf die Hallen 11 und 12 erweitert. Die Ausstellung beinhaltet nun insgesamt 18000 qm (1989: 12000 qm). Wie auch im letzten Jahr findet in den Hallen 7 bis 10 die Messe "aktiv leben '90" statt. Die Öffnungszeiten sind vom 24. bis 26. August 1990 jeweils von 10.00 bis 18.00 Uhr. Der Eintrittspreis für Erwachsene beträgt DM 7,-, für Schüler und Studenten DM 4,-. Die Preise gelten für die ATARI-Messe '90 in den Hallen 11 und 12 und gleichzeitig für die "aktiv leben '90" in den Hallen 7 bis 10. Die Preise enthalten außerdem eine Rückfahrt mit öffentlichen Verkehrsmitteln bis VRR Preisstufe 5.

E_Handel

E_Handel ist ein Programm, das für den Fahrradhandel entwickelt wurde, inzwischen aber so großen Anklang gefunden hat, daß es beispielsweise auch im Spielwarenhandel verwendet werden kann. Dabei wurde Wert auf möglichst einfache Bedienung gelegt. 5000 Artikel können mit 1 MB RAM, 10000 ab 1.5 MB RAM verwaltet werden. Dabei ist keine Festplatte erforderlich. Die Kosten für ein betriebsfertiges System mit Drucker und Software belaufen sich damit auf ca. DM 2500,- und sind daher auch für kleinste Läden erschwinglich. Der Mehrplatz-Betrieb ist möglich. Die sonst üblichen EDV-Nummern zur Identifizierung von Artikeln spielen in E_HANDEL eine Nebenrolle.

Das Suchen und Finden von Artikeln kann (muß aber nicht) ausschließlich durch Blättern und anschließendem Anklikken geschehen. Einfügungen neuer Artikel an beliebiger Stelle und Umsortierungen sind möglich. Im Registrierkassenbetrieb sind vier offene Rechnungen gleichzeitig möglich. Nicht vorhanden sind Kundenadreßdatei und Ausgangsrechnungsüberwachung, da die meisten Verkäufe im Fahrradhandel Barverkäufe darstellen. Das Programm ist also nicht für Möbelgeschäfte, Elektrogroßhandel o.ä. geeignet.

Peter Reichel
Buersche Straße 105
4500 Osnabrück
Tel. (0451) 87888

Chili, Sherlook und Cato

Chili ist eine Videografikkarte für den ATARI ST (ab Mega ST 1) mit folgenden Möglichkeiten: Echtzeitdigitizer mit 65000 Farben, Genlock (digital), VHS- und SVHS-Qualität, frei manipulierbarer Speicher mit 512 kB RAM, Farbkorrekturen, Echtzeiteffekte wie formen, spiegeln, drehen oder Trickeffekte, die das Bild auf eine Kugel legen. Diese Effekte machen die Videokarte zu einem echten Novum. Chili ist in zwei Versionen verfügbar, als Videokarte mit Trickeffekten für DM 3800,- und als reiner Echtzeit- und Echtfarbendigitizer für DM 2300,-. Beide Karten verfügen über die Genlock-Möglichkeit. Die Bedienungs-Software gehört ebenso zum Lieferumfang wie ein kleiner Titelgenerator (von Stefan Herzer). Bereits auf der CeBIT wurde Chili mit einem kompletten SVHS-System gezeigt und erstaunte die Zuschauer mit seinen komplexen Trickmöglichkeiten.

21 Punkte unterscheiden Sherlook 3.0 von ihrem Vorgänger (Version 2.3). Es fängt an beim Erkennen von Ligaturen und 8 Zeichensätzen gleichzeitig und endet bei der Anpassung an einen Großbildschirm. Der Preis der neuen Version ist DM 998,-. Ein Upgrade von Version 2.3 auf die neue Version Sherlook 3.0 ist für die Differenz der beiden Produkte erhältlich. An zusätzlichen Scanner-Treibern wird gearbeitet (z.B. Microtek).

Neu und exklusiv im Händlervertrieb gibt es bei Richter einen besonderen Leckerbissen für Börseninteressierte: CATO, eine Options- und Terminmarktanalyse-Software. Cato bietet 60 Strategien, die die sehr schnelle und genaue Berechnung aller wichtigen Daten und die grafische Darstellung zuläßt, der direkte Ausdruck als Brokerauftrag rundet das professionelle Paket ab (ein Test ist in Vorbereitung).

H. Richter Distributor
Hagener Straße 65
5820 Gevelsberg
Tel. (02332) 2706

T.I.M.-News

Aus wirtschaftlichen Erwägungen werden die Preise für die T.I.M.-Produkte eine Preisanpassung an den MS-DOS-Markt erfahren. Dadurch haben jetzt alle Produkte für ST und MS-DOS gleiche Versionsnummern und somit identische Dateiformate. Bis auf Banktransfer 1.1 wurden Preisänderungen nur an neuen Versionen vorgenommen.

TiM I: Es sind 150 Buchungen pro Periode und 12 Perioden pro Jahr möglich. Es gibt 6 Geld-, 50 Gegenkonten und 2 Steuersätze. Das Ausgabemenü umfaßt nur Kontenrahmen, -salden, Journal, Gegen- und Geldkonten. TiM I ist mit anderen TIM-Produkten NICHT datenkompatibel. TiM I kostet DM 149,-. Es gibt keine Update- oder Upgrade-Möglichkeit auf andere Produkte.

TiM IIc: TiM 1.2 wird durch TiM IIc, eine Compact-Version von TiM II, ersetzt. TiM IIc besitzt das komplette TiM II-Handbuch und basiert auch darauf. Gegenüber TiM II fehlen TiM IIc folgende Funktionen: Bilanz; GuV; Extern-Menü; Firmendaten; Sortierung. Die wichtigsten Verbesserungen: direktes Umbuchen; Benennung der Gegenkonten beim Buchen; Anzeigen des Geldkontosaldos beim Buchen; Sichern auf Knopfdruck; Listenkopf beim Ausdruck; Verbuchung des Privatanteils; Netto-Eingabe des Betrags. TiM IIc kostet DM 498,-. TiM V1.2 ST und TiM V1.1 MS-DOS werden nicht mehr angeboten. Tim I ist ein neues Low-End-Produkt. TiM IIc ersetzt TiM 1.1/1.2.

Cashflow 1.1: Die Änderungen umfassen folgende Punkte: eine neue Editzeile, in der die Eintragsnummer gezeigt wird. Bei Konto und Kürzel wird die Textbeschreibung und bei USt. der Steuersatz angezeigt. In den Hilfsfenstern kann mit Tasten gescrollt werden, bei deren Aufruf wird an den ersten Eintrag gesprungen, der größer oder gleich dem bereits eingegebenen ist. Mit TAB und Shift-TAB wird der Eintrag gewählt, das Fenster geschlossen und die nächste Kassenblattspalte angesprungen. Im Menüpunkt Suchen kann jertzt auch => und <= gewählt werden. Ab sofort erfolgt die volle Unterstützung aller Großbildschirme. Auf vielfachen Wunsch werden nun auch selektierte Einträge saldiert. Cashflow 1.1 kostet DM 498,-.

C.A.S.H. GmbH
Robert-Bosch-Straße 20a
8900 Augsburg
Tel. (0821) 703856

Repro Studio ST

Das auf der CeBIT '90 erfolgreich vorgestellte s/w-Lithografieprogramm Repro Studio ST wird ab Anfang Juni in der Version 1.1 ausgeliefert. Entscheidende Verbesserungen wurden in der Benutzerführung und in der direkten Scanner-Ansteuerung vorgenommen. Es werden zur Zeit über zehn Treiber vom Profigerät mit 256 Graustufen bis zum Handscanner angeboten. Die Umwandlung von s/w in bis zu 64 Graustufen ermöglicht auch Profiarbeit mit den einfachsten Scannern. Für Großbildschirme werden keine gesonderten Treiber benötigt, da das Programm sauber unter GEM programmiert wurde. Die Version 1.1 ist nun auch mit einem Vektorisierungsmodul erhältlich. Das Update wird allen registrierten Anwendern angeboten und ist erhältlich bei:

Trade IT
Odenwaldstraße 36
6102 Pfungstadt
Tel. (06157) 86430

Preise im Wert von über DM 15.000,- zu gewinnen!



2

6

Offizielle Leserumfrage ´90

Es ist wieder einmal soweit. Um die ST-Computer nicht an Ihnen "vorbeizuschreiben", benötigen wir Ihre Meinung. Wie könnten wir sie besser erfahren, als es mit einer Leserumfrage möglich ist? Damit wir möglichst viele Zuschriften erhalten, haben wir unsere Umfrage auch für Sie attraktiv gestaltet. Wie bei unserer letzten Umfrage im November '88 gilt es auch dieses Mal wieder, viele Preise zu gewinnen. Folgende Preise im Gesamtwert von weit über DM 15.000,- warten auf ihre neuen Besitzer (die Hersteller sind alphabetisch geordnet):

Application Systems ///, Heidelberg

- *1x Script:* Signums kleiner Bruder. Das Programm eignet sich hervorragend zum Schreiben von Briefen, Anleitungen, Arbeiten und jeglicher Korrespondenz. *(Bild 1)*
- *1x Esprit:* Das verflixte Spiel mit und um Kugeln. Vorsicht: Das Esprit-Fieber greift um sich und wird auch Sie nicht verschonen!

Eickmann Computer, Frankfurt

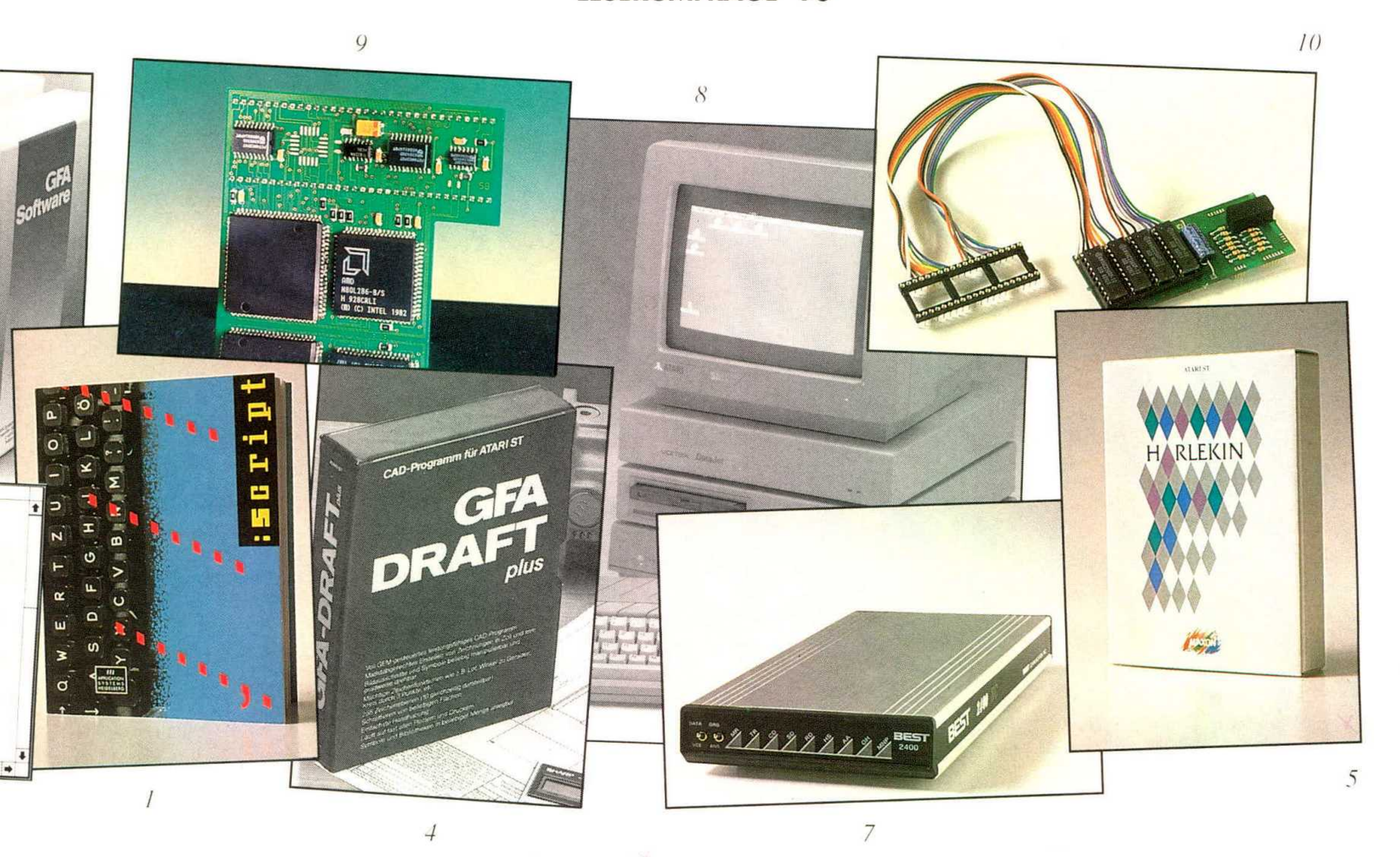
- *1x EM 124:* Dieser Graustufen-Multisync im ATARI-Look ist genau der richtige Monitor für alle, die Kabelwirrwarr und Platzverschwendung nicht mögen. Der Monitor stellt alle Auflösungen des ST problemlos dar. *(Bild 2)*
- *1x EX-60 L:* Diese leise Festplatte ist allen Anwendern ein Begriff. Mit 60 Megabyte ist sie auch für gehobene Ansprüche nicht zu klein. Mitgeliefert wird der weithin bekannte Eickmann-Festplattentreiber, um komfortabel mit der schnellen, leisen Platte arbeiten zu können. *(Bild 2)*

GFA Systemtechnik, Düsseldorf

- *1x GFA-BASSIC EWS 3.5:* Mit Interpreter und Compiler ist das Entwicklungssystem komplett ausgerüstet. Programme sind schnell und einfach zu schreiben und können sofort in direkt ausführbare Programme gewandelt werden.
- *1x GFA-Assembler:* ein professioneller Makro-Assembler für engagierte 68000-Programmierer: leistungsfähiger Editor mit integriertem Assembler und Linker sowie nachladbarem Debugger. *(Bild 3)*
- *1x Gem Utility Package:* Programmierer von GEM-Anwendungen werden sich für diesen Preis interessieren. Bibliotheken für jeden Zweck sind zahlreich vorhanden.
- *1x GFA-Strukto:* Strukto ist eine dialogorientierte programmierte Unterweisung zum strukturierten Programmieren. Verlassen Sie den Spaghetti-Code und lernen Sie, Ihre Programme strukturiert aufzubauen!
- *1x GFA-Draft-plus V3.1:* ein zweidimensionales CAD-Programm, das jetzt durch Spline-Funktionen, Metafile-Treiber und DXF-Konverter erweitert worden ist. *(Bild 4)*
- *1x GFA-BASIC-Grafik-und Sound-Bibliothek:* Verschönern Sie Ihre GFA-BASIC-Programme mit Grafik- und Sound-Routinen, die Ihr Auge verwöhnen werden. Einige hundert einlinkbare Prozeduren machen das Programmieren zum Kinderspiel.

MAXON Computer, Eschborn

- *5x Harlekin:* Harlekin ist ein Multi-Accessory. Ob Sie nun einen Editor, eine Datenbank, einen Taschenrechner, ein Terminal-Programm, eine ASCII-Tabelle, einen neuen Zeichensatz, eine RAM-Disk brauchen; dies alles und noch viel mehr bietet Ihnen Harlekin in einem Programm! *(Bild 5)*



9 8 10 1 4 7 5

Shift, Flensburg

- *10x Arabesque:* Arabesque bietet Ihnen die Möglichkeit, sowohl mit Raster- als auch mit Vektorgrafiken zu arbeiten. Oder beides zu vereinen. Arabesque unterstützt alle wichtigen Grafikformate. *(Bild 6)*
- *10x Themadat:* Aus tausenden von Daten diejenigen zu finden, die man wirklich braucht, erfordert eine leistungsfähige Datenbank. Themadat orientiert sich an Begriffen, die den ganzen Datensatz beschreiben. Verweise, Grafiken, Editoren, Themadat ist einfach komplett.

Telekommunikation Kaben Riis (TKR), Kiel

- *10x Multiterm pro:* Bildschirmtext in 32 Farben für alle ATARI ST-Rechner: endlich ist es möglich! Natürlich kann Multiterm pro auch auf einem monochromen Monitor mit Graustufendarstellung benutzt werden.
- *1x BEST 2400 L:* Ein bewährtes Modem mit den Übertragungsgeschwindigkeiten 300, 1200 und 2400 Baud. Das Modem ist sowohl für SysOps als auch für jeden DFÜ-Freak bestens geeignet. Steigen Sie in die faszinierende Welt der DFÜ ein! *(Bild 7)*

Vortex Computersysteme, Flein

- *1x DataJetX40:* Der DateJetX40 ist ein Festplatten-Subsystem der schnellen Klasse: Eine eingebaute SCSI-Platte überzeugt durch Geschwindigkeit, die mitgelieferte Software durch Komfort. Mit 40 Megabyte ist das Gerät auch für größere Ansprüche geeignet. *(Bild 8)*
- *3x ATonce:* Durch ATonce werden ST bzw. Mega ST zu einem AT-kompatiblen Computer mit 8 MHz Taktfrequenz. Der Norton-Faktor beträgt 6,5; das sind zirka 50% mehr als ein vergleichbares System mit V30-8 MHz-Prozessor! 704 kB DOS-Speicher, 24 GEMDOS-kompatible Partitionen und noch viel mehr warten auf die neuen Besitzer! *(Bild 9)*

Weide Elektronik, Hilden

- *1x 512 kByte RAM-Expansion:* Für 520 ST-Besitzer kommt die Erleichterung: Rüsten Sie Ihren Rechner ohne Löten auf 1 MB auf! Die RAM-Karte wird einfach eingesteckt und ist sofort lauffähig - natürlich voll bestückt. *(Bild 10)*

Wir bedanken uns bei allen Spendern für die Preise und hoffen auf eine rege Beteiligung. Bitte bedenken Sie, daß wir die ST-Computer nur dann Ihren Wünchen anpassen können, wenn möglichst viele Meinungen bei uns eingehen. Bitte beachten Sie, daß Mehrfacheinsendungen erstens das Ergebnis verfälschen und zweitens zwecklos sind, da sie aussortiert werden.

Wie funktioniert's?

Um an der Verlosung teilzunehmen, füllen Sie bitte die beigeheftete Postkarte aus. Zur Not oder bei Platzmangel auf der Karte können Sie natürlich die Bögen aus dem Heft benutzen (heraustrennen oder kopieren). Die ausgefüllte Postkarte senden Sie bitte bis zum 31.7.1990 an:

MAXON Computer GmbH
Stichwort: Leserumfrage '90
Industriestraße 26
D-6236 Eschborn

Es gilt das Datum des Poststempels. Der Rechtsweg ist, wie immer, ausgeschlossen. Mitarbeiter der MAXON Computer GmbH und deren Angehörige dürfen nicht an der Verlosung teilnehmen. Nun brauchen Sie nur noch etwas Glück, um einen der Preise bald auf Ihrem Schreibtisch stehen zu haben.

MP

Leserumfrage '90

40 Fragen beantworten und gewinnen! *(Bitte nutzen Sie die im Heft befindliche Karte)*

1. Welches ST-Modell besitzen Sie?

☐ 260 ☐ 520 ☒ 1040 ☐ Mega1 ☐ 2 ☐ 4 ☐ STE

2. Betriebssystemversion? ☐ 1.0 ☒ 1.2 ☐ 1.4 ☐ 1.6

3. Wieviel Speicher (MB)? ☐ 0.5 ☒ 1 ☐ 2 ☐ 4 ☐ ____

4. Welche Monitore? ☐ Farbe ☒ s/w ☐ geplant ____

5. Welche Massenspeicher besitzen Sie?

☒ Laufwerk(e) ☐ Festplatte ☐ Andere ☒ geplant

MB u. Firma: MB u. Firma: MB u. Firma: MB u. Firma:

40 MB Hard & Soft

6: Drucker? ☐ 8-/9-Nadeln ☒ 24-Nadeln ☐ Laser

☐ geplant von ☐ ATARI ☐ ____

7. Welche internen / externen Erweiterungen benutzen Sie?

☐ Grafikkarte ☐ 16 MHz-Erweiterung ☐ Coprozessor

☐ Netzwerk ☐ Modem ☐ Emulator ☒ andere ☐ geplant

Scanner

Wie groß ist Ihr Interesse an:	sehr groß	groß	mäßig	klein	Note
8. Listings	☐	☒	☐	☐	3
9. Anwenderprogrammen	☐	☒	☐	☐	2
10. Anwendungsbeispielen	☒	☐	☐	☐	2
11. Hardware	☒	☐	☐	☐	3
12. Hardware-Einzeltests	☒	☐	☐	☐	1
13. Selbstbauprojekten	☒	☐	☐	☐	4
14. Grafik	☐	☐	☒	☐	3
15. Musik	☐	☐	☐	☒	2
16. Spielen	☐	☐	☐	☒	1
17. News	☒	☐	☐	☐	1
18. Aktuellen Berichten	☒	☐	☐	☐	2
19. Quicktips	☒	☐	☐	☐	3
20. Datenfernübertragung	☐	☒	☐	☐	3
21. Marktübersichten	☒	☐	☐	☐	4
22. Public Domain	☐	☒	☐	☐	2
23. Demo-Disketten	☐	☐	☒	☐	2
24. Sonderdisketten	☐	☒	☐	☐	3

25: Bevorzugte Programmiersprachen? (Interpreter/Compiler)

C, ~~Assembler~~, Fortran

26: Welche Programmiersprache würden Sie gerne lernen?

Pascal

27. Wieviele Zeilen sollte ein Listing höchstens haben?

400

28. Wünschen Sie Selbstbauprojekte? ☒ ja ☐ nein

29. Welcher Erfahrungsgrad sollte für 28. vorausgesetzt werden? ☐ Anfänger ☒ Semi ☐ Profi

30. In welcher Preisklasse (DM) sollte ein Projekt (28.) liegen? ca. DM 200,-

31. Haben Sie Interesse sich mit Beiträgen an der ST-Computer zu beteiligen? ☐ ja ☒ nein

32. Wie nutzen Sie Ihren Computer?

☒ privat ☒ geschäftlich

33. Wie schätzen Sie Ihre ST-Kenntnisse ein?

☐ sehr hoch ☐ hoch ☐ mittel ☒ gering ☐ keine

34. Kaufen Sie die ST-Computer?

☐ Abonnement ☒ immer ☐ oft ☐ manchmal ☐ nie

35. Lesen Sie die ST-Computer?

☒ immer ☐ oft ☐ manchmal ☐ nie

36. Wie finden Sie die Gestaltung des

Innenteils? ☐ sehr gut ☒ gut ☐ mäßig ☐ schlecht

des Titels? ☒ sehr gut ☐ gut ☐ mäßig ☐ schlecht

37. Welche Themen wünschen Sie sich für die Zukunft?

Finanzmathematische Programme, Assembler Tips / Tricks, Marktübersicht Programmiersprachen

38. Welche weiteren ComputerZeitschriften lesen Sie?

c't, Computer aktuell

39. Wieviel Geld geben Sie monatlich für Ihr Hobby aus?

DM 100,-

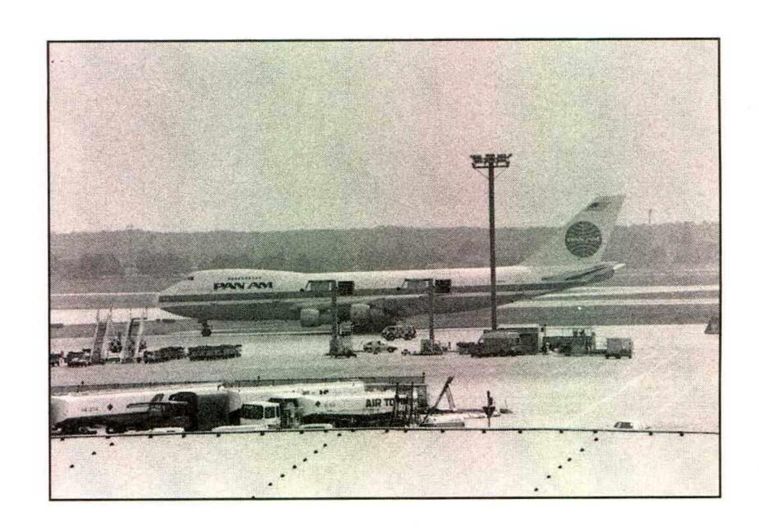
40. Haben Sie Interesse an weiteren ATARI-Produkten?

☐ in ST-Computer ☒ generell

	besitze ich	ja	nein
Portfolio	☐	☒	☐
TT	☐	☒	☐
PC	☐	☐	☒
Lynx	☐	☒	☐

Flugsicherung mit dem ST

Unsicherer Luftraum?



Vorsicht, Ironie: Sie alle wissen, wie zuverlässig der ATARI ST ist. Eben dieses zuverlässige Gerät, mit dem die meisten von Ihnen häufig arbeiten dürften, wird auch zur Flugsicherung in der Luftraumnutzungszentrale Frankfurt eingesetzt. Wie gut, daß es noch die Deutsche Bundesbahn gibt...

Genug der Ironie. Die Aufgabe, die der ST zu bewältigen hat, ist gar nicht so schwierig. Wesentlich schwieriger ist es da schon, Ihnen diese Aufgabe zu erklären. Sicherlich werden Sie sich vorrangig dafür interessieren, warum eine solche Aufgabe mit einem ATARI ST erledigt wird, obwohl der sogenannte "Industriestandard" dem Gerät von allen Seiten zusetzt. Die Antwort ist eine leichte welche: Ein Bediensteter der Luftraumnutzungszentrale (LRNZ) hatte den Wunsch, die konventionelle Arbeit des Fernschreibens schneller, bedienungsfreundlicher und leichter zu gestalten. Was lag also näher, als einen ST für diese Anwendung einzusetzen? So hielt der ST, der auf unserem Heimatplaneten nur einmal für eine solche Aufgabe eingesetzt wird, Einzug in die Flugsicherung.

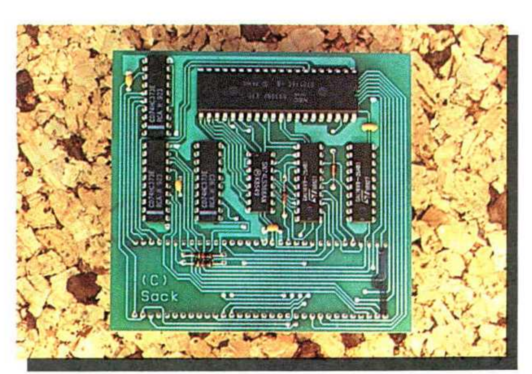
Bevor Sie jetzt kein Flugzeug mehr besteigen, lassen Sie sich erst die Aufgabe des ST erklären. Nehmen Sie an, ein dikker Nebel liegt über Frankfurt und begrenzt die Sicht auf 1,5 km. Dabei können natürlich weniger Flugzeuge starten bzw. landen, als es normalerweise möglich und üblich wäre. So könnte beispielsweise eine Beschränkung lauten, daß aus Paris nur noch ein Flugzeug pro 20 Minuten landen darf. Solche Nachrichten müssen dann natürlich so schnell wie möglich zu ihrem Bestimmungsort, in diesem Fall Paris, gelangen. Das entsprechende Datennetz nennt sich AFTN (Aeronautical Fixed Telecommunication Network) und ist ein Flugsicherungsnetz, das auf jedem größeren Flughafen der Erde installiert ist. Wie in fast allen Bereichen existiert auch hier ein konkurrierendes Netz. Es nennt sich SITA (Societé International de Télécommunication Aeronautique) und wurde errichtet von verschiedenen großen Fluggesellschaften. Mit diesen beiden Flugsicherungsnetzen kann der ATARI ST der LRNZ kommunizieren.

Bitte sehen Sie in Abbildung 1. "Rembrandt?" werden Sie sagen. Richtig, genau das habe ich auch im ersten Moment gedacht. Die Flugsicherungsanstalten der verschiedenen Länder arbeiten mit weltweit genormten Abkürzungen, da die Länge einer solchen Nachricht 1800 Bytes nicht überschreiten darf. Damit auch Sie den Inhalt der (fiktiven) Nachricht verstehen, schreite ich hiermit offiziell zur Erklärung. In der ersten Zeile wird der Absender der Nachricht eingetragen. Die folgenden sechs Zeilen geben die Empfänger der Nachricht an, es sind verschiedene Länder und Städte. EDDF steht dabei zum Beispiel für Frankfurt. In den folgenden zehn Zeilen, die mit Nummern von 1 bis 9 versehen sind, steht endlich die eigentliche Nachricht. Zeile 1 sagt aus, welche Nummer das Schreiben (EXECUTION NR.###) hat und an welchem Tag es gültig ist (VALID ######). Damit das Schreiben wiedergefunden werden kann, besitzt es am Ende der Zeile ein Aktenzeichen (LE##). Die nächste Zeile erklärt dem Empfänger, wieso sie zu befolgen ist: "DUE TO RESTRICTIONS BY LECB". LECB ist dabei die Abkürzung für Barcelona. Bei unserem Beispiel hat also Barcelona Einschränkungen für den zu erwartenden Flugverkehr gegeben. Die dritte Zeile zeigt dem Empfänger an, von wann bis wann es gültig ist (PERIOD CONCERNED: #### - #### UTC). Die Abkürzung UTC steht dabei für "Universal Time Coordinated" und ist die Zeit, die in Greenwich gemessen wird. Die angegebene Zeit ist in diesem Fall die Zeit, in der die betroffenen Flugzeuge sich über dem Einflugpunkt Frankreich befinden (ETO FRENCH ENTRY POINTS); ETO heißt dabei "Estimated Time Over". Grämen Sie sich nicht, wenn Sie schon jetzt nicht mehr alles verstehen: Zum Schluß erkläre ich alles noch einmal auf hochdeutsch. Unterdessen sollten Sie sich jedoch Zeile vier betrachten, denn diese Zeile gibt an, wer überhaupt von der Nachricht betroffen ist. In diesem Fall sind das Flugzeuge mit den Zielen Palma (LEPA), Ibiza (LEIB) und Mahon (LEMH). TFC ist dabei die Abkürzung für "Traffic". Eventuell könnten nur bestimmte Flughöhen von den Restriktionen betroffen sein, deshalb werden auch diese spezifiziert. In Zeile fünf sehen Sie bereits, daß alle Flughöhen (FL = Flight Level) davon betroffen sind. Die nächste Zeile ist etwas unverständlich. SLAP und ATFMU sind die Abkürzungen für "Slot Allocation Procedure" (=Zeitrasterzuweisungsverfahren) und "Air Traffic Flow Management Unit"

Die *MS-DOS*-Welt erschließen mit PC-Speed ...

PC Speed für den ST

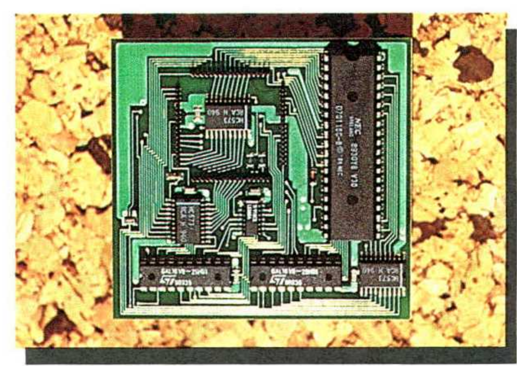
Über 20 000 Anwender haben weltweit in den ersten neun Monaten nach seiner Einführung PC-Speed gekauft und sich mit diesem Spitzenprodukt Zugang zu den Tausenden von MS-DOS Programmen verschafft.



PC-Speed erweitert den ATARI ST zum MS-DOS-Rechner

PC Speed für den STE

Obwohl PC-Speed für den STE erst seit vier Wochen auf dem Markt ist, haben schon über 1 000 Benutzer zugegriffen und integrieren MS-DOS Programme wie selbstverständlich in ihre Computerarbeit.



PC-Speed erweitert den ATARI STE zum MS-DOS-Rechner

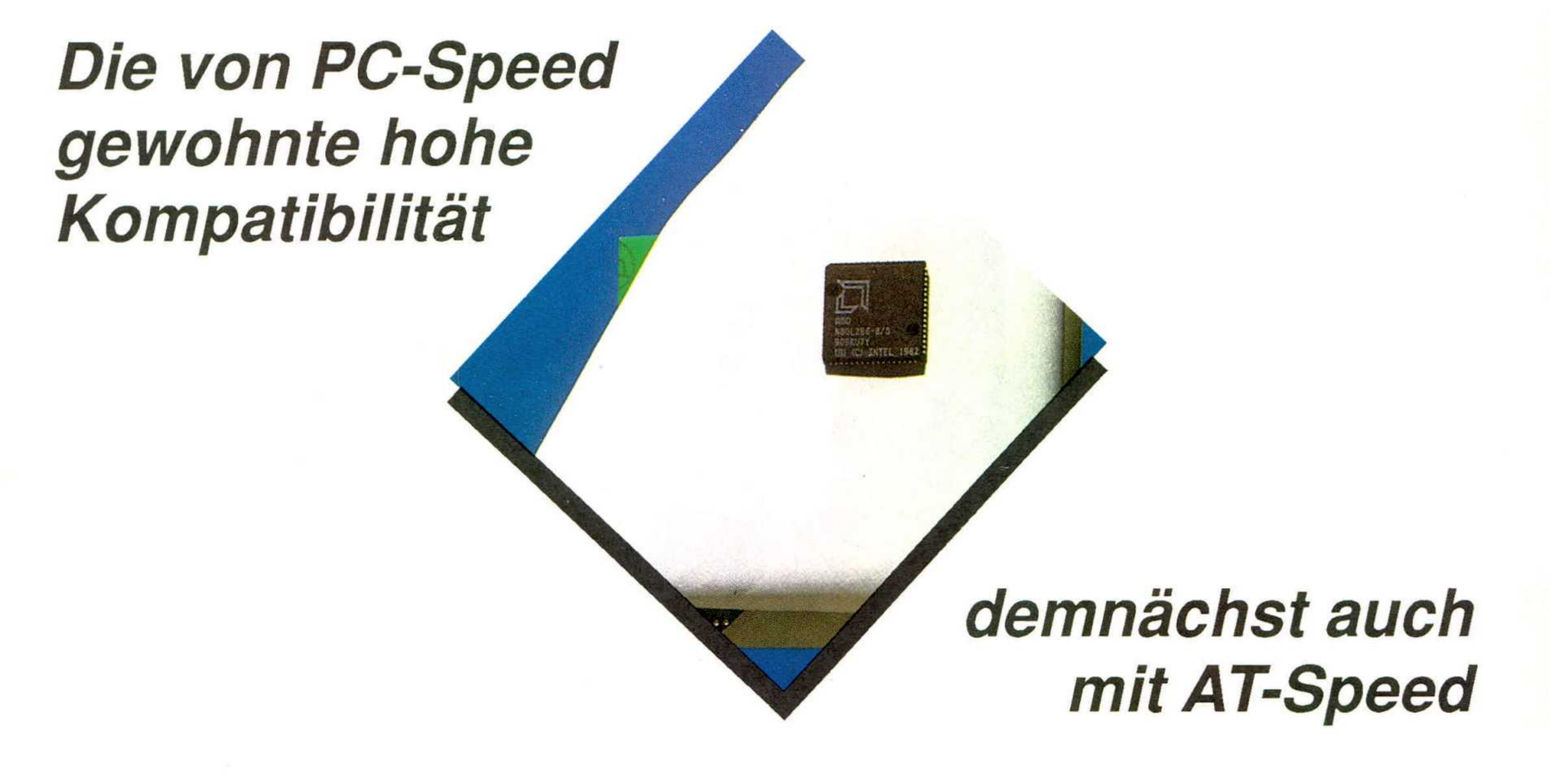
Was hat PC-Speed diesen Erfolg gebracht?

- die modernste Technologie
- die saubere Lösung
- die sichere Funktion
- die hohe Leistung
- die HighTech-Qualität
- die Verfügbarkeit im Bundesgebiet bei über 400 Fachhändlern
- die gute Hotline-Betreuung
- der Update-Service

Mit PC-Speed profitieren ST- und STE-Anwender von der Leistung ihres Atari-Computers und gleichzeitig von der unübertroffenen Programm-Vielfalt der MS-DOS Welt

...die Überraschung

AT Speed



Die von PC-Speed gewohnte hohe Kompatibilität

demnächst auch mit AT-Speed

Die Erfahrung des Entwicklers H. J. Sack macht es möglich!

* MS-DOS ist eingetragenes Warenzeichen der Microsoft Corp.

Vertrieb weltweit:

Heim Verlag

Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Bundesrepublik:
bei über 400 ATARI-Händlern

Österreich:
Darius
Inh. K. Hebein
Hartlebengasse 1-17/55
A-1220 Wien
Tel.: 00 43 - 222 - 239580

Schweiz:
Data Trade AG
Landstr. 1
CH-5415 Rieden-Baden

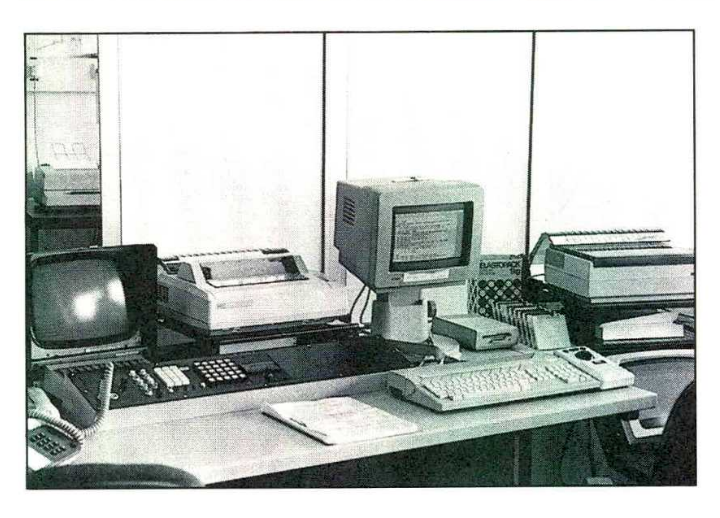
für alle anderen Länder:
COMPO SOFTWARE GmbH
Postfach 1051
D-5540 Prüm (FRG)
Tel.: 0 65 51 / 62 66

(=LRNZ bzw. Luftraumnutzungszentrale). Soll im Klartext heißen: Das Zeitrasterzuweisungsverfahren wurde von der LRNZ Frankfurt bestimmt. Oder ganz anders: Die Luftraumnutzungszentrale Frankfurt hat bestimmt, wieviele Flugzeuge für diese Einschränkung in einem bestimmten Zeitraum geschickt werden dürfen. Es ist einleuchtend, daß ein solcher Satz abgekürzt werden muß. Kurz: "Rate: Slap Atfmu Frankfurt". Sollte sich ein Empfänger dieser Nachricht angesprochen fühlen, kann er Rückfrageaktionen starten. Dazu existiert Punkt 7. "Für Zeitrasteranfragen benutzen Sie bitte den Code: Balearic Islands" (SRQ heißt "Slot Request" gleich Zeitrasteranfrage). Wer also ein Abflugfenster beantragen möchte, muß sich mit dem Code "Balearische Inseln" melden. Punkt acht schränkt die gesamte Nachricht ein weiteres Mal ein, denn hier steht, daß Entlastungsstrecken bekanntgegeben werden, sobald sie verfügbar sind. Sollten also Umwegstrecken geflogen werden, während die einschränkende Nachricht gültig ist, werden sie auf dem gleichen Wege bekanntgemacht. Punkt neun ist freigehalten für verschiedene Nachrichten. In unserem Fall steht hier die Nachricht "Achtung Zürich: Wenn Sie über Italien fliegen - Zeitrasterzuweisungsverfahren Luftverwaltungscenter". Die Abkürzungen: "LSZH" = Zürich, "SLAP" hatten wir schon, "ATMC" = Air Traffic Management Center. "NNNN" beendet, wie auch bei Telexen, die Nachricht.

```
ZCZC LZA000
FF EKDKZDZK LKPRZDZX LYBBZDZX LOVVZDZX LSAZZDZX EDDFZDZX
FF EDZZZDZX EDZZZGZX EDDFYPYX EBBDEUCX
FF EDCCNWXX EDAZYOYX EDDFZDZX
FF EDDFSITE EDDFAALO EDDFPAAO EDDFTWAO
FF EDDFSITG
FF EDDFSITK
000000 EDDFZDZX
1. FLOW CONTROL EXECUTION NR.### VALID ######       // (LE##)
2. DUE TO RESTRICTIONS IMPOSED BY LECB
3. PERIOD CONCERNED: #### - #### UTC -ETO FRENCH ENTRY POINTS-
4. TFC CONCERNED: TFC DEST. LEPA/LEIB/LEMH
5. FL CONCERNED: ALL
6. RATE: SLAP ATFMU FRANKFURT
7. COMMUNICATION: COM #       (SEE INF. MSG)
   -FOR SRQ USE CODE: #BALEARIC ISLANDS
8. OFF LOAD ROUTE #WILL BE OFFERED WHEN AVAILABLE
9. REMARKS: ATTN LSZH: IF VIA ITALY - SLAP ATMC
NNNN
```



Abb. 1: Ein Wust von Abkürzungen wird vom ST in die Welt geschickt.



Wer hätte das gedacht? In Frankfurt sichert ein ATARI ST den Luftraum

Wie bitte?

Für den Fall, daß Sie die Nachricht nicht verstanden haben, noch einmal im Klartext: "Flugkontrollausführungsmeldungsnummer ###, gültig am ###### // Aktenzeichen ##. Gesendet aufgrund von Einschränkungen aus Barcelona. Betroffene Zeit: #### bis #### Greenwich-Zeit - geschätzte Zeit über dem Einflugpunkt Frankreich. Betroffener Verkehr: Verkehr mit dem Ziel Palma, Ibiza und Mahon. Betroffene Flughöhen: alle. Zeitrasterzuweisverfahren wird bestimmt durch die Luftraumnutzungszentrale Frankfurt. Kommunikationsnummer: COM # (siehe Info-Nachricht) - Für Zeitrasteranfragen bitte den Code 'Balearische Inseln' benutzen. Entlastungsstrecken werden angeboten, sobald sie verfügbar sind. Bemerkung: Achtung Zürich: Flüge über Italien bitte mit der Luftraumnutzungszentrale koordinieren!"

Nun sollte hoffentlich klar sein, was die ominöse Nachricht in Abbildung 1 zu bedeuten hat. Eben solche Nachrichten werden in Frankfurt mit dem ST erstellt und in alle Welt verschickt. Wo Sie die Doppelkreuze finden, werden normalerweise natürlich Daten eingesetzt, etwa das aktuelle Datum oder verschiedene Kommentare. Dafür beinhaltet das LRNZ-Programm einen einfachen Texteditor. Innerhalb der letzten Jahre haben die Mitarbeiter der Flugsicherung genügend Erfahrungen gesammelt, um jetzt sagen zu können, alle wichtigen Nachrichten, die normalerweise auftreten können, auf Diskette gesammelt zu haben. So wird, wie in unserem Beispiel, der Ansturm auf die Balearen im Sommer natürlich größer sein als zu einem anderen Termin. In London ist oft Nebel, in vielen Städten wird öfter gestreikt als in anderen. All die Nachrichten, die oft benötigt werden, sind auf Disketten verfügbar und dadurch innerhalb von 2 Minuten an ihrem Bestimmungsort - immerhin spielen bereits wenige Minuten eine sehr große Rolle bei der Flugsicherung: Ab einem bestimmten Zeitpunkt ist die Kapazität eines Flughafens erreicht. Einen Stau kann es in der Luft nur sehr schwierig geben, also müssen Flugzeuge mit entsprechendem Ziel später oder gar nicht starten.

Reserven

Interessant ist, daß das System aus einem 520 ST mit 512 kB, einem Diskettenlaufwerk SF 314 und einem SM 124-Monitor besteht. Eine Festplatte wird man vergeblich suchen, da die Nachrichten nicht so übermäßig lang sind, daß man sie auf einem riesigen Medium speichern müßte.

Wirft das Gerät Bomben oder verabschiedet sich gänzlich (dieser Fall ist jedoch bisher noch nicht vorgekommen), stürzen nicht etwa Flugzeuge ab oder können nicht mehr starten, sondern innerhalb einer Minute wird ein Ersatzgerät aufgestellt, um den korrekten Betrieb zu gewährleisten. Mit dem Ersatzgerät arbeitet zur Zeit die Sekretärin, die mit 1st_Wordplus und Calamus Briefe schreibt. Für dieses System stehen selbstredend auch eine Festplatte und ein SLM 804-Laserdrucker zur Verfügung. Auch die Korrespondenz der LRNZ wird also über einen ST abgewickelt.

Guten Flug

Sie sehen auch in diesem Anwendungsgebiet ein weiteres Beispiel für den professionellen Einsatz des ATARI ST. In diesem Fall bestimmt er zwar nicht gerade das Leben von Menschen, wird jedoch dafür eingesetzt, die Sicherheit im Flugverkehr zu gewährleisten. Bleibt nur noch, Ihnen einen guten Flug zu wünschen...

MP/CPL

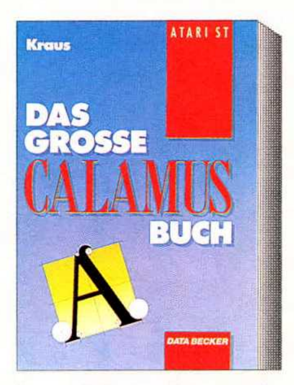
STARKE ST-BÜCHER IM JUNI!

DAS KOMPLETTE NACHSCHLAGE-WERK ZUM GFA-BASIC!

Lassen Sie sich von einem Profi beraten, wie Sie Ihre GFA-BASIC-Programme rundherum professionell gestalten können. Ziehen Sie bei Ihrer Programmierung das große GFA-BASIC-Buch zu Rate. Denn hier finden Sie alles für eine fortgeschrittene Programmierung: die praktische Anwendung der einzelnen GFA-Befehle der neuen Version 3.5, das Einbinden von System-Routinen, alles Wissenswerte zur Programmstruktur, beispielhafte Grafik- und GEM-Programmierung – alles immer anhand praktischer, anschaulicher Beispiele erklärt. Selbstverständlich erhalten Sie mit diesem Buch eine komplette Beschreibung aller Optionen des 3.5-Compilers und einen ausführlichen Anhang – unter anderem mit einer Übersicht sämtlicher Fehlermeldungen. Auf einer mitgelieferten Diskette finden Sie zusätzlich noch zahlreiche beispielhafte Programme. Das große GFA-BASIC-Buch – ein unverzichtbares Nachschlagewerk mit dem Know-how eines Profis. Dieses Buch wird Ihre Karriere als Programmierer jederzeit zuverlässig begleiten.

Litzkendorf
ATARI ST
Das große GFA BASIC 3.5 Buch
Diskette im Buch
Interpreter und Compiler
Umfangreicher Einsteigerteil sowie eine umfassende Beschreibung aller Befehle und Funktionen von GFA BASIC bis einschließlich Version 3.5
DATA BECKER

Litzkendorf
Das große GFA-BASIC-Buch
Hardcover, inklusive Diskette,
899 Seiten, DM 59,-
ISBN 3-89011-363-X



Das große Buch zu Calamus zeigt, was es heißt, DTP-Experte zu sein. So erfahren Sie z.B., wie Rahmen „numerisch" bearbeitet oder mehrere Rahmen zusammengefaßt werden, wie Rahmen für gedrehten Text entstehen, wie man Vektor- und Rastergrafiken exportiert bzw. importiert und welche Schrifteffekte es gibt. Mit vielen wichtigen Hinweisen zur Installation, zur Benutzerführung, zu Drucker und Scanner. In einem großen Praxisteil finden Sie zusätzlich noch zahlreiche Anwendungsbeispiele, die Ihnen die Leistungsfähigkeit des Programms demonstrieren und Ihnen gleichzeitig ein sicheres Gespür für eine ansprechende Gestaltung geben.

Kraus
Das große Calamus-Buch
392 Seiten, DM 39,-
ISBN 3-89011-346-X

SO SOLLTE IHR HANDBUCH ZUM ATARI AUSSEHEN.

Auch wenn Sie mit Ihrem ST grundsätzlich zurechtkommen, es gibt immer wieder einmal ein Problem, das sich ohne weiteres nicht lösen läßt. Mal ist es die Frage nach dem Erstellen einer RAM-Disk, mal funktioniert beim Booten etwas nicht, oder aber der Drucker will nicht so, wie Sie wollen. Mit dem großen ST-Handbuch lösen Sie all diese Probleme sozusagen im Handumdrehen. Einfach nachschlagen, und schon wissen Sie, worauf es ankommt. Bei Problemen mit dem Desktop genauso wie bei der Tastatur oder den Schnittstellen. Dabei werden Sie natürlich noch jede Menge interessanter Neuigkeiten über Ihren Rechner erfahren und auf eine Reihe nützlicher Tricks stoßen. Zahlreiche Tips zur Pflege Ihres Rechners und zum „Rechner-Tuning" gibt Ihnen dieses Buch genauso weiter wie viele, kleine Reparatur-Hinweise. Ein Nachschlagewerk, das Sie immer wieder einmal benutzen werden – nicht zuletzt dank seiner klaren Gliederung. Und das auch zukünftig aktuell ist: Der ST/E wird natürlich ebenfalls behandelt.



Liesert
Das große ST-Handbuch
Hardcover, 377 Seiten
DM 49,-
ISBN 3-89011-273-0

Kraus
ATARI ST
DAS GROSSE BUCH ZU 1ST WORD PLUS
DATA BECKER

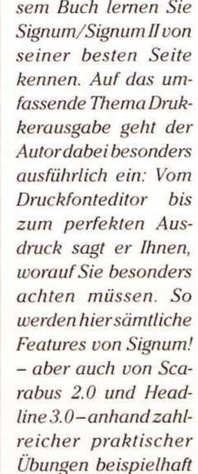
Für alle Viel- und Schönschreiber: Das große Buch zu 1st Word Plus – die detaillierte Anleitung zu allen Funktionen Ihres Textprogramms (auch die der aktuellen Version 3). Dabei beschreibt der Autor in erster Linie den Einsatz von 1st Word Plus im praktischen Alltag, denn rund die Hälfte des Bandes ist den praktischen Anwendungen gewidmet – von kleinen zu umfangreicheren Texten, von Tabellen zur Grafikeinbindung. Und zu fast allen Bereichen gibt's noch viele nützliche Tips und Tricks. Das große Buch zu 1st Word Plus – auch mit einer Erklärung der Zusatzprogramme 1st XTRA, 1st Proportional und des Profi-Text-Moduls.

Kraus
Das große Buch
zu 1st Word Plus
314 Seiten, DM 39,-
ISBN 3-89011-347-8

SIGNUM IM GRIFF: ZWISCHEN TEXT UND DTP.

Signum wird gerne als das Zwischenglied zwischen Textverarbeitung und DTP-Programm bezeichnet – eine Charakterisierung, die zeigt, wie leistungsstark diese Software sein muß. Mit dem großen SIGNUM!-Buch haben Sie das Zeug, diese Textverarbeitung voll auszunutzen. Ob Diskettenoperationen, Texteinstellungen, Fußnotenverwaltung, Spaltensatz, Druckausgabe oder Bildoperationen – mit diesem Buch lernen Sie Signum/Signum II von seiner besten Seite kennen. Auf das umfassende Thema Druckerausgabe geht der Autor dabei besonders ausführlich ein: Vom Druckfonteditor bis zum perfekten Ausdruck sagt er Ihnen, worauf Sie besonders achten müssen. So werden hier sämtliche Features von Signum! – aber auch von Scarabus 2.0 und Headline 3.0 – anhand zahlreicher praktischer Übungen beispielhaft erklärt. Das große SIGNUM!-Buch – als praxisorientierter Einstieg genauso gut geeignet wie als umfassendes Nachschlagewerk. Ihre Texte verdienen es, gut auszusehen – nutzen Sie Signum in Verbindung mit diesem Buch.





Kraus
Das große SIGNUM!-Buch
346 Seiten, DM 39,-
ISBN 3-89011-313-3

TOS 1.4: WAS IST NEU?

Was ist neu an TOS 1.4, und wie können Anwender und Programmierer davon profitieren? Das Update-Buch erklärt die neuen Funktionen des Betriebssystems anhand von zahlreichen praxisorientierten Beispiel-Programmen in den Sprachen GFA-/Omikron-Basic, Turbo C und Assembler. Mit allen interessanten Neuigkeiten für Anwender und für Programmierer. Außerdem ein Kapitel über bekannte Fehler und ein ausführlicher Anhang. Die beste Einführung zu Ihrem TOS-Update (auch für den neuen ATARI ST/E).



Pauly
Das TOS-1.4-Update-Buch
205 Seiten, DM 29,-
ISBN 3-89011-256-0

SOFORT BESTELLEN...

...bei DATA BECKER, Merowingerstraße 30, 4000 Düsseldorf 1

Bezahlung ☐ per Nachnahme.
☐ mit beiliegendem Verrechnungsscheck.
(Zzgl. DM 5,- Versandkosten, unabh. von der best. Stückzahl)

Bestellung:

Vorname/Name

Straße/Nr.

PLZ/Ort

DATA BECKER

Konstruktive Resourcen

Ein Vergleichstest von Resource Construction Sets

Eines der wichtigsten Utilities beim Erstellen von GEM-Programmen ist ohne Zweifel das Resource Construction Set. Dieses schmucke Programm, das dazu dient, auf möglichst einfache Art Dialogboxen und Menüleisten zu erstellen, und ursprünglich von Digital Research als Teil des Entwicklungspakets herausgebracht wurde, hat inzwischen ein paar Geschwister und einen Nachfolger bekommen, die vorgeben, es mit ihm aufnehmen zu können oder besser zu sein. Dies ist sicherlich ein Grund, diese unterschiedlichen Resource Construction Sets einmal unter die Lupe zu nehmen.

Ein Adventure im Jahre 1985

Als ich im Oktober 1985 voller Stolz vor meinem 520er (natürlich damals noch ohne Plus und 512 kB zusätzlich) saß und mir anschaute, was er alles zu bieten hatte, beschäftigte ich mich mit den wenigen Dingen, die ich zu diesem Zeitpunkt besaß und das war nicht mehr als das Entwicklungspaket. Am meisten fiel dabei ein Programm auf, das den vielsagenden Namen RCS besaß. Einmal ganz abgesehen davon, daß ich natürlich keinerlei Ahnung hatte, wie man mit GEM umzugehen vermag, brachte ich auch mit viel Engagement innerhalb des Programms nicht viel zusammen, zumal ich auch nicht so genau wußte, für was das RCS überhaupt gut war. Also habe ich daraufhin die drei blauen Telefonbücher (will sagen die mitgelieferte auf den ATARI angepaßte MS-DOS-GEM-Dokumentation) gewälzt, und bin zu der Erkenntnis gekommen, daß dort auch nicht viel drin steht. Mit vereinten Kräften mehrerer Leute haben wir es dann doch geschafft, eine Dialogbox mit dem RCS zu erstellen (was bedeutete, daß uns sogar inzwischen klar geworden war, was der Sinn des Programms eigentlich ist...). Als besonderer Erfolg wurde von uns einige Wochen später die Erkenntnis gefeiert, daß die '.H'-Datei (die sogenannte Definitionsdatei) tatsächlich einen Sinn macht und die Programmierung von GEM-Programmen erheblich erleichterte. Das ganze kam uns vor wie ein Adventure, bei dem man ab zu mal wieder etwas Neues entdeckt...

Szenenwechsel

Glücklicherweise arbeiteten immer mehr Leute mit dem ST und benutzten somit auch ausgiebig das Entwicklungspaket mit seinem RCS. Dadurch wurde der Umgang mit dem RCS zur Selbstverständlichkeit. Dennoch hat(te) dieses Programm mehr als nur eine Macke, ganz zu schweigen von der Tatsache, daß jegliche Dokumentation fehlte (nach meiner Kenntnis ist auch heute die Beschreibung in unserem Sonderheft die einzige). Inzwischen wurden einige neue Entwicklungspakete mit den unterschiedlichsten Sprachen auf den Markt gebracht, wobei sehr schnell klar war, daß diese Pakete für die professionelle Programmentwicklung selbstverständlich auch ein Resource Construction Set enthalten mußten. Daher kam sehr bald von Megamax ein RCS heraus (das allerdings später in Deutschland vom Kuma RCS abgelöst wurde und daher auch in diesem Vergleichstest nicht aufgeführt ist), und Digital Researchs RCS 2.0 erblickte über irgendwelche dubiose Kanäle das Licht der Welt bis ATARI Deutschland auch diese Version in ihr Entwicklungspaket übernahm. Ein völlig neu gestaltetes RCS kam nach einiger Zeit von Kuma auf den Markt und seit kurzem ist das Wercs von Hisoft dabei, seinen Anspruch auf dem Software-Markt geltend zu machen. Höchste Zeit, dieses wichtige Utility in seinen verschiedenen Erscheinungsformen unter die Lupe zu nehmen. Nebenbei: Ich kann natürlich nicht garantieren, daß das alle Programme dieser Art sind, allerdings sind dies meiner Meinung die bekanntesten Vertreter. Ich möchte dabei so vorgehen, daß ich die RCS einzeln in ihrer Bedienbarkeit und ihrem Funktionenumfang beschreibe und bewerte. Danach möchte ich noch einmal speziell auf die folgenden drei Punkte im direkten Vergleich eingehen: Icon-Editor, Definitionsdatei und die Anleitung. Gehen wir also einfach chronologisch vor. Das erste RCS, das existierte, war das

ATARI RCS, Version 1.x

Sicherlich ist es Sinn und Zweck eines RCS', das Erstellen von Dialogboxen und Menüleisten zu ermöglichen, wobei das Wichtigste eines Programmierhandwerkzeuges in erster Linie die Bedienerfreundlichkeit sein sollte. Auch wenn es auch heute immer wieder Utilities gibt, an denen ein Hacker seine Freude haben würde, stehen Effizienz und leichte Handhabung an erster Stelle. Umso mehr verwundert es einen, daß das ATARI RCS schon damals eine gut durchdachte Handhabung bei der Erstellung der Bäume und Objekte zur Verfügung stellte.

Die Handhabung läßt sich am besten an Bild 1 erkennen. Nach dem Starten des RCS' ist der Bildschirm im wesentlichen in zwei Hälften geteilt (oberer Teil von Bild 1). In der oberen Hälfte findet man die Grundstrukturen der Dialogboxen, Menüleisten und Alertboxen und im unteren Bereich die vom Benutzer erstellten Objekte. Soll nun eine Dialogbox kreiert werden, wird einfach eine neue Dialogbox mit der Maus von oben nach unten gezogen, und schon fragt das RCS nach einem Namen (unter dem diese Dialogbox später im Programm angesprochen werden kann). Danach einen Doppelklick auf das Symbold dieser Dialogbox - und sie Sesam-öffnet sich. Anstelle der Grundstrukturen erscheinen nun in der oberen Box die unterschiedlichen Objekttypen (unterer Teil von Bild 1), die in einer Dialogbox sein können. Auch hier genügt wieder ein einfaches Verschieben eines Objektes vom oberen Fenster in das untere, um dieses Objekt in der Dialogbox unterzubringen. Das Edieren der Objekteigenschaften geschieht, indem man auf das entsprechende Objekt doppelklickt, worauf sich eine (meist große) Box öffnet, in der man alles einstellen kann. Diese Art der Bedienung gefällt mir auch heute noch sehr gut, allerdings mit der folgenden Einschränkung: Immer wenn man von dem oberen Fenster in das untere etwas übernehmen möchte, muß dazu das obere Fenster erst angeklickt werden. Bei etwas umfangreicheren Dialogboxen kann das Hin- und Herschalten zwischen den beiden Fenstern schon etwas nerven. Dennoch gefällt mir ganz besonders, daß die gesamten Eigenschaften eines Objektes nach dem Doppelklick in einer großen Box geändert werden können.

Schon damals bot das RCS einige ganz nützliche Funktionen (siehe Bild 2), die einem das Leben wirklich vereinfachten. So konnte man die Objekte verstecken und versteckte Objekte wieder hervorholen (HIDE/UNHIDE) und auch die Reihenfolge der Objekte durch Sortieren ändern. Dies ist besonders wichtig, wenn man umfangreiche Objektstrukturen erstellt hat. Werden diese nämlich mit ihren beispielsweise 30 Objekten gezeichnet, sieht es nicht gerade schön aus, wenn sie wild durcheinander aufgebaut werden (eine Tatsache, die unbedingt bei großen Drop-Down-Menüs beachtet werden sollte!). Ganz praktisch ist auch die Funktion 'Flatten' (Abflachen), mit der man vorher durch ein Vaterobjekt zusammengefaßte Objekte (praktisch, wenn man eine Gruppe von Objekten gleichzeitig verschieben möchte) wieder unabhängig machen kann, indem das Vaterobjekt ohne seine Kinder gelöscht werden kann. Eine Funktion, die recht wichtig ist, wenn man Resourcen erstellen möchte, die auf unterschiedlichen Auflösungen laufen sollen, ist 'Snap'. Wenn diese Funktion angewählt ist, achtet das RCS darauf, daß die Koordinaten auf Buchstabenbreiten und -höhen abgelegt werden, wodurch eine korrekte Umrechnung durch das GEM in der entsprechenden Auflösung sichergestellt ist. Praktisch ist auch, daß man dem RCS in drei Stufen mitteilen kann, wie gut man mit dem Programm und Objektstrukturen umgehen kann (einstellbare Warnstufe). Abhängig von der Einstufung gibt das RCS von sehr vielen Warnungen bei bestimmten Vorgängen bis zu keine Warnung mehr von sich. Die Einstellung 'Locked' ist beispielsweise sehr wichtig, wenn man Texte von Resourcen übersetzen möchte, ohne daß dabei die Reihenfolge der Objektnumerierung verändert wird.

Mit 'Load...' können Icons, die erstellt worden sind, in das RCS geladen werden. Womit wir bei den negativen Eigenschaften angekommen wären... Wo Licht ist, ist auch Schatten, der besonders in den ersten Versionen besonders groß war. Zunächst einmal muß festgestellt werden, daß das RCS zwar Icons, die sicherlich wichtig für anwenderfreundliche Oberflächen sind, verarbeiten kann, deren Bilder aber im RCS nicht erstellt und auch nicht verändert werden können. In der Anfangszeit gab es noch nicht mal einen (externen) Icon-Editor, so daß das Thema Icon erst einmal vergessen werden konnte. Erst eine ganze Weile später folgte dann ein Icon-Editor, über dessen Qualität ich mich lieber nicht auslassen möchte, so daß zu diesem Zeitpunkt bemerkt wurde, daß das RCS einen dicken Fehler beinhaltete: Das Abspeichern von Resourcen mit Icons klappte nicht, denn dabei stürzte das RCS erst einmal ohne Vorwarnung ab. Glücklicherweise gibt es das RCS heute in der Version 1.4, mit der auch Icons problemlos verarbeitet werden können. Trotzdem ist es ein großes Manko, die Icons außerhalb des RCS ändern zu müssen (die Grunderstellung eines Icons ist, wenn es gut sein soll, recht aufwendig und kann deshalb meines Erachtens ruhig mit einem anderen Programm geschehen).

Alles in allem ist das RCS in der Version 1.4 ein professionelles Programm, mit dem man ohne Probleme arbeiten kann, obgleich es inzwischen sehr günstig sehr günstig zu haben ist. Ich möchte allerdings nicht vergessen zu erwähnen, daß daß die Definitionsdatei (dazu unten mehr) ohne Konvertierung nicht kompatibel zum RCS auf MS-DOS-Rechnern ist.

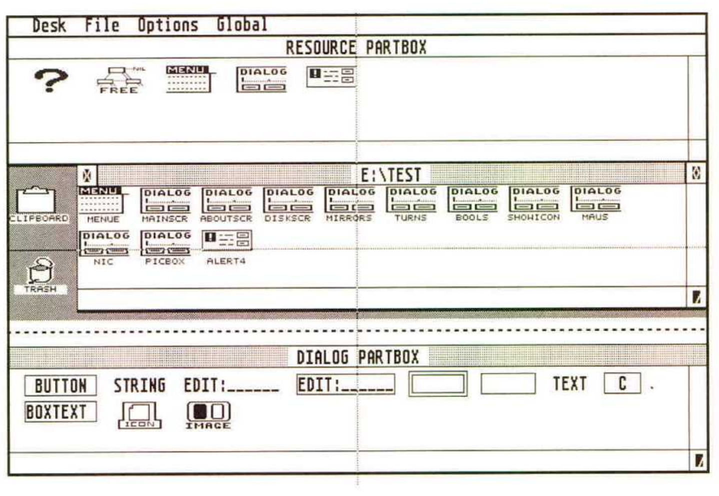


Bild 1: Die Hauptarbeitsbereiche des RCS 1.4

Desk | File | Options | Global
About the RCS 1.4 | New | Info... | Output...
Desk Accessory 1 | Open... | Name... | Safety...
Desk Accessory 2 | Merge... | Hide
Desk Accessory 3 | Close | Unhide
Desk Accessory 4 | Save | Sort...
Desk Accessory 5 | Save As... | Flatten
Desk Accessory 6 | Abandon | Snap
Quit | Load...

Bild 2: Auch das RCS 1.4 bietet einiges...

Das RCS 2.x von ATARI...

...ist die weiterentwickelte Version des RCS 1.x und wird inzwischen bei den unterschiedlichsten Entwicklungspaketen (ATARI-Entwicklungspaket für Entwickler, GFA-BASIC, SPC-Modula ...) mit ausgeliefert. Bei dieser Version sind nicht nur die Fehler der 1.x-Version ausgebügelt worden, sondern auch neue Funktionen eingebaut, die ich nicht unerwähnt lassen möchte. Dieses RCS ist auch im Entwicklungssystem des MS-DOS-GEM zu finden, so daß Dateien untereinander austauschbar sind (das bezieht sich im wesentlichen auf die Definitionsdatei). Leider kann man es meines Wissens nicht erwerben, so daß es quasi außer Konkurrenz startet.

Zunächst präsentiert sich das RCS 2.1 nach dem Laden so, wie Sie es in Bild 3 erkennen können. Besonders angenehm empfinde ich, daß es nach wie vor Fenster gibt, in dem die edierten Baumstrukturen zu finden, und ein Fenster, in dem die Ausgangsstrukturen vorhanden sind. Sehr von Vorteil ist dabei, daß man diese beiden Fensternicht mehr aktivieren muß (wie es noch im RCS 1.x der Fall war), sondern einfach einen neuen Baum aus dem unteren Fenster in das obere schiebt - ein Fortschritt in der Bedienerfreundlichkeit. Eine neue Art der Bedienung findet man aber, wenn man ein Objekt verändern möchte. Dann nämlich klappen aus den acht links oben befindlichen Icons Pop-Up-Menüs heraus, die die Einstellungen verbergen. Der Vorteil dieser Menüs ist, daß die schon fast überladenen Dialogboxen im RCS 1.x durch kleinere ersetzt werden konnten. Der Nachteil ist, daß diese Einstellungen der Dialogboxen in diesen Pop-Up-Menüs verschwinden. Der Nachteil ist folgender: Wenn Sie beispielsweise mehrere Flags eines Objektes edieren möchten, müssen Sie immer auf das dritte Icon von oben in der rechten Reihe klicken und dann das Flag anklikken, welches geändert wird. Daraufhin schließt sich allerdings das Menü wieder, was bedeutet, daß Sie für die Änderung des nächsten Flags wieder das Icon anklikken müssen, das Flag ändern und so weiter...

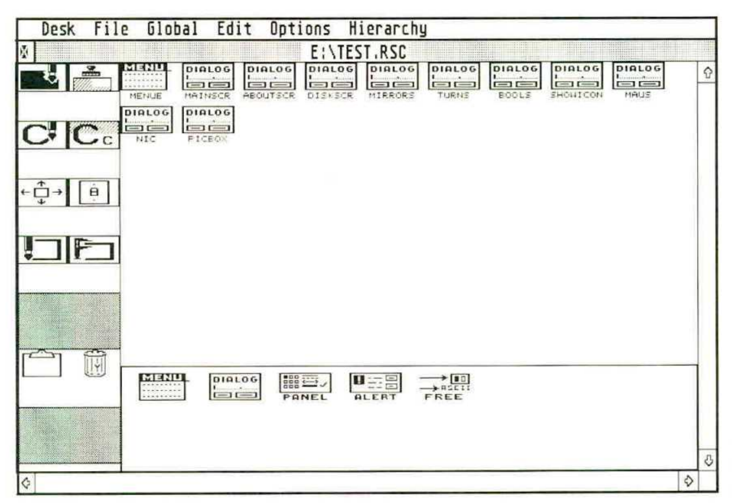


Bild 3: Der Arbeitsbildschirm des RCS 2.1

Im großen und ganzen sind die Funktionen des RCS 2.x identisch mit denen des RCS 1.x, wobei diese etwas umstrukturiert und auch teilweise ausgelagert wurden. So ist das Kopieren und Löschen eines Objektes auch über die Menüzeile möglich, da diese Funktionen bei der Verwendung der Clipboard- und Papierkorb-Icons in Zusammenhang mit großen Objekten zu Problemen in der Handhabung führen konnte. Außerdem kann man, um sehr große Dialogboxen zu verarbeiten, das linke Fenster mit den Pull-Up-Menüs sowie das untere Fenster mit seinen Baum-Grundstrukturen verschwinden lassen. Ein Pull-Up-Menü möchte ich dennoch nicht missen, da es mit seinen Funktionen die Möglichkeit gibt, Objekte innerhalb anderer Objekte auszurichten. Dabei kann das Objekt links, rechts, ober- und unterbündig sowie mittig bezüglich der Höhe und Breite des Vaterobjektes ausgerichtet werden. Außerdem kann die Größe des Objektes auf die Höhe oder Breite des Vaterobjektes gebracht werden. Ich beschreibe diese Möglichkeiten deshalb, weil ich diese Funktion sehr oft verwende und leider in keinem der anderen RCS wiederfinde. Wenn Sie zu den RCS-Anwendern gehören, werden Sie mir wahrscheinlich rechtgeben, daß das Ausrichten von Objekten nicht selten vorkommt und bei einigen Dialogboxen eine langwierige Arbeit ist, die durch die Funktionen des RCS 2.x sehr erleichtert werden kann.

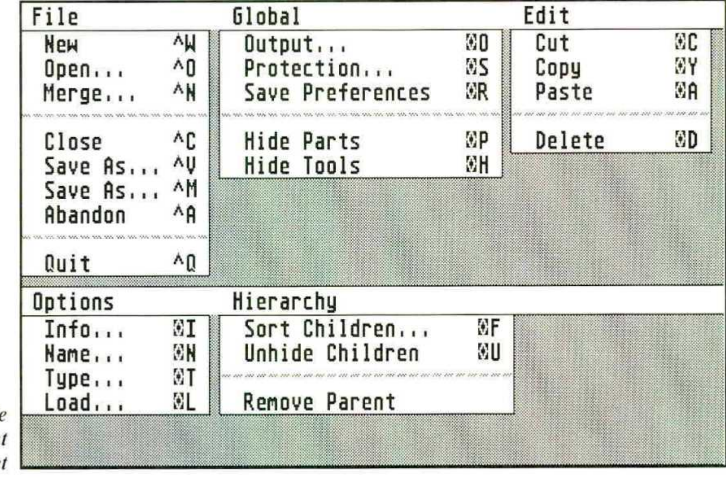


Bild 4: Viele Funktionen gut geordnet

Einige Zeit, nachdem GEM auf dem ATARI implementiert worden war, gab es eine Veröffentlichung des Digital-Research-Mitarbeiters Tim Oren (der übrigens Mitautor des RCS' ist), die sich 'Professional GEM' nannte. In einer der Folgen wurde erwähnt, daß nicht alle Bits innerhalb der Objektstruktur tatsächlich auch von GEM genutzt und - viel besser noch - ignoriert würden. Dadurch ist es dem Anwender möglich, an dieser Stelle eigene Informationen unterzubringen. Um diese Möglichkeit zu nutzen, müßte es aber ein RCS geben, in dem man diese Bits schon bei der Erstellung der Resourcen setzen kann. Dies ist mit dem RCS 2.x möglich geworden, da jedem Objekt ein sogenannter EXTENDED OBTYPE hinzugefügt werden kann. Diese Zahl wird in das obere Byte des ob_type eines Objektes geschrieben und kann später innerhalb des Programms ausgewertet werden. Leider ist dies die einzige Möglichkeit, da es, wie das Kresource zeigt, noch mehr Möglichkeiten gibt, einem Objekt an ungenutzten Stellen Informationen beizugeben.

Äußerst positiv ist mir aufgefallen, daß

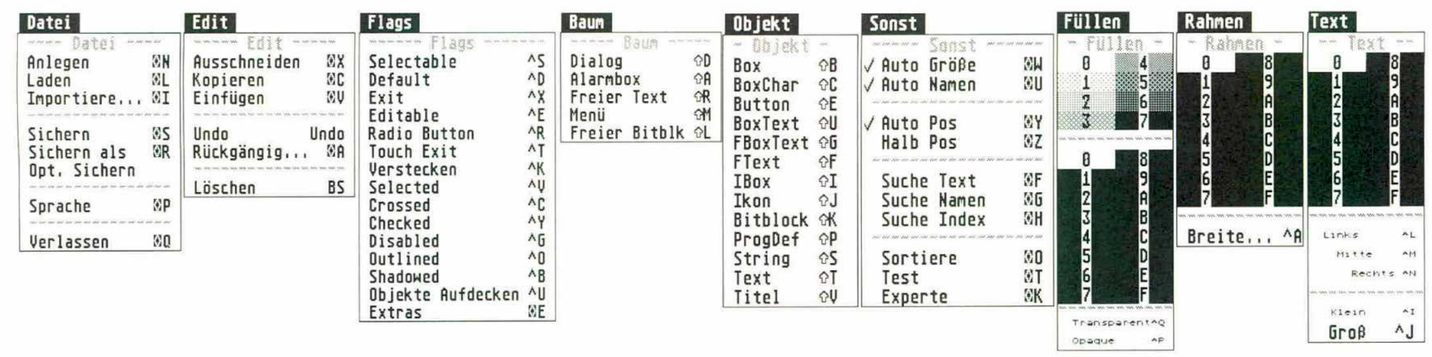


Bild 6: Die umfangreiche Menüleiste von Wercs

alle Funktionen der Drop-Down-Menüs (ok, bis auf eine, und zwar 'Remove parent', was dem alten 'Flatten' entspricht) durch die Tastatur anwählbar sind. Leider schließt das die Funktionen der meines Erachtens ganz interessanten aber letztendlich etwas umständlichen Pull-Up-Menüs nicht mit ein. Zusammengefaßt bietet das RCS 2.x gegenüber dem 1.x die einfachere Handhabung bezüglich der Fenster sowie die praktischen Funktionen zum Ausrichten eines Objektes, wohingegen das Ändern eines Objektes mit Hilfe der Pull-Up-Menüs teilweise etwas nerven kann. Die Möglichkeit 'extended obtypes' zu nutzen, spricht auch dafür, von RCS 1.x auf 2.x umzusteigen. Die Kompatibilität der DFN-Datei zu der MS-DOS-Version ist sicherlich als Vorteil für die zu sehen, die mit beiden Betriebssystem arbeiten wollen oder müssen.

Das K-Resource 2

Zwei Dinge zeigt uns der Name dieses Programms: Erstens ist der Name Resource Construction Set von Digital Research geschützt und zweitens handelt es sich offensichtlich um die zweite Version des Programms, die unter anderem deshalb nötig wurde, weil das alte Programm auf dem Blitter-TOS nicht lief.

Lädt man das K-Resource, befindet man sich auf einer Desktop-ähnlichen Bedieneroberfläche, die dadurch, daß sie sehr bekannt wirkt, sofort zum Arbeiten einlädt. Eine nette Tatsache ist, daß das K-Resource 2 die Disk-Icons und den Papierkorb genauso postiert, wie Sie sie auf dem Desktop liegen haben [kleine Anmerkung: Es benutzt vermutlich die Einstellungen, die im *Desktop.Inf* gespeichert sind - wahrscheinlich über *Shel_get()*]. Auf dieser Oberfläche ist es möglich, mehrere Inhaltsverzeichnisfenster zu öffnen, zwischen denen Dateien hin- und herkopiert werden können. Soll eine RSC-Datei bearbeitet werden, wird einfach die entsprechende Datei von dem Directory-Fenster auf die K-Resource-Desktop-Fläche gezogen und schon öffnet sich ein Fenster, in dem nun nicht mehr Dateien, sondern Baumstrukturen zu finden sind. Ein sehr großer Vorteil bei diesem Konzept ist, daß es dadurch auch möglich ist, mehrere RSC-Dateien gleichzeitig zu öffnen, zu bearbeiten und zwischen ihnen auch Baumstrukturen auszutauschen. Dadurch können beispielsweise Objekte, die oft benötigt werden und mit viel Mühe erstellt worden sind (beispielsweise Icons), in einer speziellen Datei gesammelt und später als Baumstruktur oder auch nur als Objekt wieder hinzugeladen werden.

Sicherlich ist K-Resource als das erste wirklich vollständige RCS anzusehen, da es zum ersten Mal auch einen Icon-Editor integriert bekommen hat (siehe unten). Im wesentlichen enthält das K-Resource allerdings die gleichen Funktionen wie das RCS von Digital Research, nur mit dem Unterschied, daß sie mehr von der Menüzeile auf das Desktop verlegt worden sind. Daher ist es das RCS für den Mausliebhaber. Praktisch ist auch, daß die Objektnummer des Objektes, das unter der Maus zu finden ist, immer sofort in der Menüzeile (ein seltsamer Ort...) eingeblendet wird, wodurch man sicherstellen kann, daß die Objekte in einer Reihenfolge liegen, wie man sie auch wirklich haben möchte. Eine schöne Art der Handhabung sind Menüs, die am Objekt erscheinen, wenn man dieses anklickt (siehe Bild 5 rechts oben). Damit ist die Wahl der möglichen Funktionen sehr gut dargestellt und vom Benutzer anwählbar - kein lästiges Herumsuchen nach 'Was kann man nun mit diesem Objekt tun'-Funktionen innerhalb der Menüleiste. Meines Erachtens ist damit der vielstrapazierte Begriff 'information hiding' in bezug auf die Bedienerfreundlichkeit relativ gut in die Tat umgesetzt worden. Man sollte sich also nicht durch die etwas kleine Menüzeile täuschen lassen.

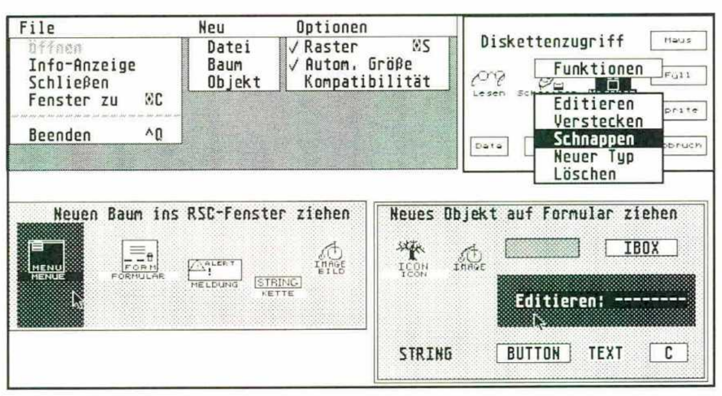


Bild 5: Ein Teil der Auswahlmöglichkeiten von K-Resource 2

Glücklicherweise findet man die schönen, großen, vom RCS 1.x bekannten Dialogboxen, in denen man alle Einstellungen ohne Drop-Down-Menüs machen kann, im K-Resource 2 wieder. Hinzugekommen sind die (beim RCS 2.x) erwähnten 'extended obtypes', aber auch 'extended flags' und ein Feld für 'extended status', wobei Flags, wie es sich so gehört, tatsächlich bitweise gesetzt werden können. Alle diese Werte, die im K-Resource 2 gesetzt werden, können später im Resource in den High-bytes der Objektstruktur-Elemente ob_type, ob_flags und ob_state wiedergefunden und verarbeitet werden, woraus sich eine hohe Flexibilität für den Anwender ergibt. Als erstes RCS bietet das K-Resource 2 auch die Möglichkeit, eine Baumstruktur einmal auszuprobieren, um beispielsweise zu erkennen, ob alle Objekte (in der Art) selektierbar sind (wie man es sich vorgestellt hat). Angenehm fiel mir dabei auf, daß hierbei vorher geprüft wird, ob überhaupt ein Ausgang für diese Dialogbox besteht, da ein fehlender Ausgang zum Hängenbleiben in der Testfunktion führen würde (einen solchen

Test gibt es bei Wercs auch, das ebenso vorher prüft, ob ein Ausgang vorhanden ist).

Insgesamt stellt sich dieses Resource Construction Set als komplettes System dar, das zwar noch kleine Wünsche offen läßt (siehe auch unten über den Icon-Editor sowie Definitionsdatei und Anleitung), das aber durch seine Bedienerfreundlichkeit gestattet, relativ schnell Resourcen zu erstellen.

Hi-Soft WERCS...

...das ursprünglich für das FTL-Modula gedacht war, welches in der ST-Computer 3/90 getestet worden ist, gibt es seit ein paar Monaten über CCD auch einzeln zu kaufen. Dies erklärt wahrscheinlich auch, warum es (noch) nicht ganz so bekannt ist wie die anderen RCS.

Das Wercs, dessen Name mich seltsamerweise an Arbeit erinnert (Works...), ist angetreten, den Kampf gegen das arbeitsintensive Gestalten der Resourcen zu bestehen. Nach dem Laden findet man zunächst einmal nur ein leeres Fenster und eine riesige Menüzeile vor. Im Gegensatz zum K-Resource 2 ist es nicht möglich, mehr als eine Resource-Datei zu bearbeiten. Nur durch einen Trick über Importieren und Einfügen kann man das erste Objekt innerhalb des ersten Baumes einer zweiten Resource-Datei (am besten den Teil des Satzes noch einmal langsam lesen) in die in der Arbeit befindliche Datei importieren. Bedenkt man, daß schon das RCS von Digital Research eine Funktion zum Einfügen anderer Resourcen hat, so finde ich es etwas bedauerlich, daß dies in Wercs nicht vorhanden ist. Ich gebe allerdings zu, daß ich diese Funktion erst ab dem Momemt vermißte, als ich versucht habe, Icons hinzuzuladen - solange hatte ich die Merge-Funktion im RCS nie bewußt wahrgenommen.

Wercs bietet an Funktionen (fast) alles, was man sich wünschen kann. Eine Tatsache, die man an der Unmenge von Menüeinträgen auch schon erahnen kann. Die Vermutung, daß Wercs eine andere Konzeption bezüglich der Bedienung im Vergleich zum K-Resource 2 verfolgt, ist auch sehr schnell erkennbar. Während K-Resource die entsprechenden Funktionen erst beim Anklicken des Objektes als Untermenü hervorbringt und später eventuell noch eine Dialogbox als weiteres Auswahlmenü, so setzt Wercs auf eine Menüleiste, deren Drop-Down-Menüs abhängig von dem in Bearbeitung befindlichen Objekt anwählbar sind. Nachteilig an Drop-Down-Menüs (oder auch Pull-Up-Menüs) ist, daß mehrfache Einstellungen, wie sie beispielsweise beim Ändern des Objektstatus' oder der Objektflags vorkommen, ein wiederholtes Anwählen der Drop-Down-Menüs erfordern. Dennoch braucht man dies bei Wercs nicht unbedingt, da jeder(!) Funktionseintrag in den Menüleisten von Wercs über die Tastatur zu bedienen ist. Leider fehlt eine Übersicht der Menüleisten (einer der wirklich wenigen Kritikpunkte an der sonst recht guten Anleitung), wie ich sie Ihnen in Bild 6 dargestellt habe. Hat man eine solche Übersicht neben dem Rechner hängen, kann man über Tastatur sehr schnell alle Eigenschaften einstellen und bei Bedarf noch einmal in der Menüleiste nachschauen, wie das Objekt momentan definiert ist. Wenn man also einmal die gebräuchlisten Tastenfunktionen parat hat, ist sicher ein sehr schnelles Arbeiten mit Wercs möglich.

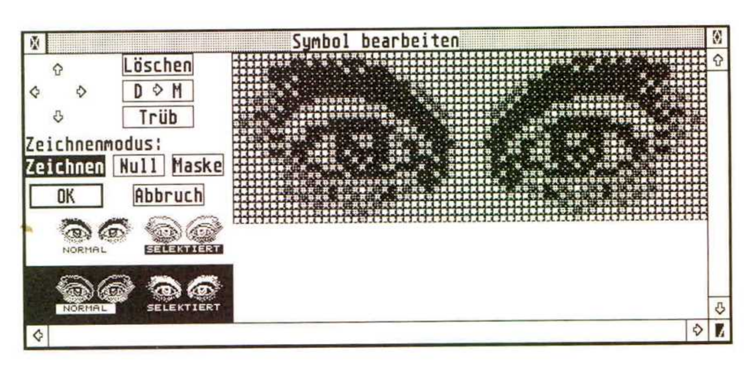


Bild 7: Ein versuchter Icon-Editor im K-Resource 2

Eine Funktion, die ich in den anderen RCS noch nicht gefunden habe, ist das Suchen nach einem definierten Namen oder Index eines Objekts oder einem Text innerhalb eines Text-/String-Objekttyps. Bei größeren Werken kann es schon von Vorteil sein, mal zu schauen, wo welches Objekt mit welchem Namen oder welcher Nummer zu finden ist, oder ob man einen bestimmten Text schon einmal benutzt hat, oder man weiß eventuell nicht mehr, wo das Objekt ist, aber daß in ihm ein bestimmter Text vorkommt...Leider fehlt in Wercs wie auch im K-Resource 2 die Möglichkeit, ein Objekt ausrichten (zentrieren etc.) zu können.

Wie man es seit dem RCS 2.x gewohnt ist, gibt es auch die Möglichkeit, einem Objekt einen 'extended obtype' zu geben. Leider findet man die im K-Resource zu findenden 'extended ob_flags' und 'extended ob_states' nicht (wobei ich mir zugegebenermaßen nicht ganz sicher bin, ob die High-Bytes von ob_state und ob_flags von Digital Research offiziell als verwendbar freigegeben worden sind und eine Veränderung dieser Bits nicht eventuell bei späteren Versionen des GEM zu Problemen führen könnte). Bei der Anzahl der Objekttypen kann man sich allerdings nicht beschweren: Es ist sogar der Objekttyp PROGDEF vorhanden, der es ermöglicht, an dieser Stelle später im Programm eigene Routinen ausführen zu lassen, und der auch eine eigene Art der Darstellung in Wercs besitzt (man sollte nur darauf achten, dieses Objekt später im Programm auch zu vervollständigen, das heißt, auch einen Einsprung in eine Routine mit unterzubringen). Als besonderes EXTRA (so heißt der Menüpunkt) ist das direkte Verändern durch Zahleneingabe der Objektkoordinaten und Größe sowie die Nummer des Nachfolgers hinzugekommen, wodurch eine Sortierreihenfolge manuell festlegbar ist. Das Sortieren, welches wie bei allen anderen RCS vorhanden ist, möchte ich erwähnen, weil man in Wercs das Sortieren nicht nur nach der Objektposition, sondern auch alphabetisch (bezüglich des Inhalts von Strings) ausführen kann (das Sortieren bezieht sich natürlich nur auf den Objektindex, der für die Reihenfolge des Bildschirmaufbaus wichtig ist, und bewirkt keine Veränderung der Objektkoordinaten).

Zusätzlich zu Wercs befinden sich auf der Diskette noch zwei kleine Programme. Mit dem einen lassen sich die Definitionsdateien anderer Resourcen (siehe unten) in das Wercs-eigene Format umsetzen. Mit dem anderen Programm WIMAGE ist es möglich, einen Bildausschnitt mit einer maximalen Größe von 128x128 Pixel aus einem Farb- oder Monochrombild auszuschneiden und in einen Bit-Block oder ein Icon umzusetzen, der/das in das Resource in Wercs über Importiere/Einfügen geladen werden kann. Für alle Wercs-Anwender und sonstige Neugierige sei hier angemerkt, daß die Ausgabedatei von WIMAGE nichts anderes als eine kleine Resource-Datei ist, die als erstes Objekt eine Box und als zweites den Bit-Block enthält!

Das Wercs ist sicherlich kein RCS, das nur für das FTL-Modula seine Berechtigung hat, sondern es kann sich durchaus mit den anderen RCS messen. Positiv fallen der große Funktionsumfang und die umfangreiche Tastaturbedienung auf.

Ein trauriges Thema...

...ist der Icon-Editor. Bedenkt man, daß sich das GEM nicht nur durch seine tollen Fensterchen und Menüleisten auszeichnet, sondern auch durch seine Piktogramme (angelsächsisch Icons), so ist es als peinlich zu erachten, daß es ATARI oder besser gesagt Digital Research bis heute nicht geschafft hat, ein Resource Construction Set herauszubringen, daß einen Icon-Editor integriert hat. Soweit mir bekannt ist, liefert die Firma ATARI bei ihrem Entwicklungspaket ein Programm mit, das sich Icon-Editor schimpft, über das ich aber lieber kein Wort verlieren möchte, da es eher zum Abgewöhnen ist (ich entschuldige mich für die harten Worte, aber ein Programm, das einem die Arbeit erleichtern soll, mit dem man sich aber eher herumquält, kann ich nicht akzeptieren).

Glücklicherweise beinhaltete das K-Resource 2 einen Icon-Editor, der annähernd seinen Namen verdient (Bild 7). Dennoch finde ich die Arbeitsgeschwindigkeit dieser Zugabe nicht gerade berauschend und die Methode die Daten und Maske in einem Bild bearbeiten zu müssen, nicht gerade richtungsweisend (erkennen Sie in Bild 7 die schwarz gezeichneten Daten und die grau dargestellte Maske?). Leider wurde die Wichtigkeit eines guten Icon-Editors im K-Resource meines Erachtens unterschätzt, so daß man (nach Aussagen auch anderer Anwender) sich gerne verleiten läßt, entweder das Icon nicht ganz so aufwendig zu gestalten oder es eventuell sogar ganz wegzulassen. Schon besser ist die Implementierung des Icon-Editors in Wercs (Bild 8), dessen Geschwindigkeit annehmbar und mit dem es auch möglich ist, die Maske und die Daten getrennt (sogar in unterschiedlicher Größendarstellung) zu bearbeiten. Selbst eine Funktion zum Zeichnen einer Linie oder das Spiegeln der Daten (allerdings nur bezogen auf das ganze Icon) ist vorhanden. Einige Leser werden wissen, daß ich den auf einer Sonderdisk erhältlichen Icon-Design geschrieben habe (mit dem es möglich ist, komfortabel außerhalb eines RCS' Icons zu erstellen und in alle der vier RCS zu laden) und diesen damit hochspielen möchte. Mir ist klar, daß ein so umfangreiches Programm wie Icon-Design nicht unbedingt in diesem Umfang in einem Resource Construction Set Platz finden muß, allerdings ärgere ich mich dennoch darüber, daß bestimmte Funktionen (wie hier ein Icon-Editor) durch ihre halbherzige Implementierung eher zum Nichtbenutzen als zum Benutzen verführen. Weder K-Resource noch Wercs bieten einen offiziellen Weg, Icons nachzuladen oder abzuspeichern (die Digital Research RCS können Icons laden). Beim K-Resource geht es nur dadurch, daß man eine zweite Resource-Datei lädt, in der man Icons gesammelt hat, und in Wercs geht es über den Trick des Importierens/Einfügens, vorausgesetzt das Icon befindet sich an zweiter Stelle in einer Resource-Datei. CCD hat allerdings in einem Gespräch erwähnt, daß sie wahrscheinlich das ICON-DESIGN-Format zum Laden und Speichern von Icons in WERCS implementieren werden (nachfragen!).

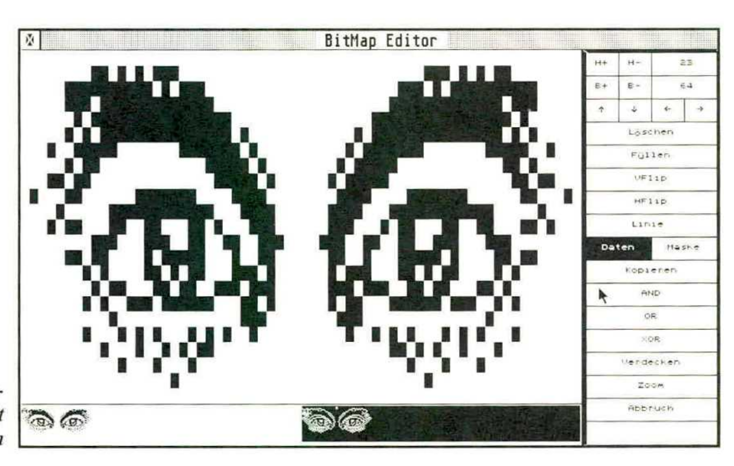


Bild 8: Mit dem Icon-Editor von Wercs läßt sich arbeiten

Die Definitionsdateien

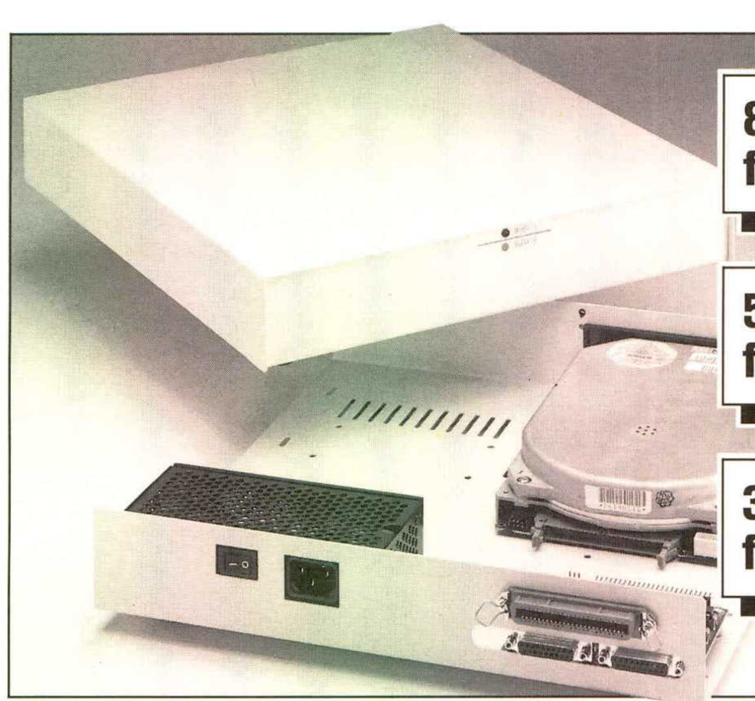
Wer von Ihnen schon einmal mit einem Resource Construction Set gearbeitet hat, weiß, daß man jedem Objekt einen Namen geben kann. Diese Namen und die Art der Baumstruktur werden allerdings nicht mit in der Resource-(RSC)-Datei, sondern in einer speziellen Datei abgespeichert. Beim RCS 1.x hat sie die Namenserweiterung DEF, die beim RCS 2.x in DFN geändert wurde, da DEF schon in Modula als Include-Dateierweiterung vergeben war und eine Änderung bezüglich der MS-DOS vorgenommen wurde (prinzipiell werden nur die Hi- und Lo-Byte-Zeiger innerhalb der Datei umgedreht). Das erste Kuma-Resource kam scheinbar vor dem RCS 2.x heraus und hat auch aus Modula-Gründen die DEF-Erweiterung in RSD umbenannt, wobei eine RCS 1.x-Datei kompatibel zu einer RSD-Datei ist. Wercs wiederum hat wieder ein anderes Dateiformat, das sich nach außen hin durch die Erweiterung HRD auszeichnet. Verwirrt genug? Sicherlich: Die eigentliche Resource-Datei, die das GEM benötigt, können alle vier Programme lesen, aber die Definitionsdateien nicht. Netterweise liefern ATARI (für RCS 1.x auf RCS 2.x) und HI-Soft (für RCS 1.x, RCS 2.x und K-Resource auf Wercs) Programme, mit denen man Fremdformate in das eigene Umwandeln kann, aber keines dieser Programme kann dies wieder rückgängig machen - man verschließt sich also immer den Weg zurück. Wer hat schon Lust, wenn er von einem RCS auf das andere wechselt, alle hundertfünfzig Namen einer Resource-Datei neu einzugeben (das Angeben der Baumart könnte man ja noch verschmerzen)? Wahrscheinlich werde ich, um diese Misere zu beheben, in einer der nächsten ST-Ecken ein Konvertierungsprogramm für alle drei Formate vorstellen (DEF->DFN und DFN->DEF funktionieren schon), das man dann auch auf der Icon-Design-Disk finden wird.

Anleitung

Leider ist das ein Thema, das allzu oft von Software-Häusern sehr 'stiefmütterlich' gehandhabt wird. So gibt es keine offizielle Anleitung zum RCS 1.x (nur eine im Sonderheft der ST-Computer), und auch das RCS 2.x verfügt auf dem ATARI über keine Anleitung, die von ATARI oder Digital Research erstellt worden ist. (Vielleicht sollte man mal die Beschreibung des GEM-Entwicklungssystems auf dem IBM danach durchstöbern.) Soweit mir bekannt ist, wird aber dem SPC Modula, welches, wie erwähnt, das RCS 2.x enthält, eine Anleitung beigelegt, die ich aber leider nicht bewerten kann, da ich sie nicht besitze. Dennoch möchte ich ein paar Worte über die Anleitungen von K-Resource 2 und Wercs verlieren, da diese immerhin als eigenständige Programme verkauft werden. Es sei noch vorausgeschickt, daß ich keine Einführung in die GEM-AES-Programmierung erwarte.

Die Anleitung des K-Resource 2 ist mit seinen 47 Seiten, die Inhaltsverzeichnis, Index und ein Listing beinhalten, relativ dünn, was aber durch das dicke, kartonartige Papier wieder etwas ausgeglichen wird... leider sind auch einige der Bilder

SCSI-Festplatten zu »Schotten-Preisen«!



Zum Beispiel:

85 MB SCSI-Festplatte (28 ms) für nur DM 1.398,–

50 MB SCSI-Festplatte (40 ms) für nur DM 1.198,–

30 MB SCSI-Festplatte (40 ms) für nur DM 998,–

Unsere SCSI-Festplatten werden komplett anschlußfertig incl. Software und Kabel ausgeliefert.

Ausstattung und Leistungsmerkmale unserer Festplatten:

- Datentransferraten >600 KByte/s (mit CDC- und Maxtorlaufwerken bis zu 850 KByte/s erzielbar), mittlere Zugriffszeiten bis zu 14 ms
- Spitzensoftware: 255 Partitionen installierbar, Passwortfunktion, jede Partition autobootfähig, Interleave 1:1 einstellbar, Cache, Backup, Optimizer in der Software enthalten
- 100% Atari-kompatibel, sämtliche Fremdbetriebssysteme (PC-Speed, PC-Ditto, Spectre, Aladin, Minix, OS-9, RTOS) sind voll lauffähig
- Superleise (3,5"-Festplatten ohne Lüfter, 5,25"-Festplatten mit thermogeregeltem Lüfter)
- Durchgeschleifter gepufferter DMA-Bus, Autoparkfunktion hardwaremäßig
- Herausgeführter SCSI-Bus (50-poliger Centronics-Anschluß, Apple MacIntosh und PC's anschließbar)
- Zweite SCSI-Festplatte im Gehäuse nachrüstbar (SCSI-Hostadapter und Gehäuse für interne zweite Festplatte vorbereitet)
- Unsere SCSI-Festplatten werden komplett anschlußfertig im Gehäuse incl. Netz-, DMA-Kabel, Software und Handbuch geliefert
- Baugleich mit LACOM

Preise:

32 MB, 40 ms, ST138N-0	DM 998,–
40 MB, 19 ms, Quantum	DM 1.298,–
49 MB, 40 ms, ST157N-0	DM 1.198,–
85 MB, 28 ms, ST296N	DM 1.398,–
80 MB, 24 ms, ST1096N	DM 1.498,–
80 MB, 19 ms, Quantum	DM 1.798,–
170 MB, 28 ms, 2x ST296N	DM 2.498,–
280 MB, 17 ms, Maxtor	DM 3.498,–
380 MB, 17 ms, Maxtor	DM 3.998,–
702 MB, 14 ms, CDC	DM 5.998,–
1200 MB, 14 ms, CDC	DM 11.998,–
44 MB, 25 ms, SQ 555	DM 1.898,–

SCSI-Kits (Festplatte und SCSI-Hostadapter für ST):

32 MB Kit (ST138N-0)	DM 848,–
40 MB Kit (P40S)	DM 1.098,–
49 MB Kit (ST157N-0)	DM 998,–
85 MB Kit (ST296N)	DM 1.198,–
80 MB Kit (ST1096N)	DM 1.298,–
80 MB Kit (P80S)	DM 1.598,–

SCSI-Hostadapter (incl. Software und DMA-Kabel)	DM 198,–
DMA-Kabel	DM 39,–
SCSI-Kabel	DM 39,–
Netzteil 50 W	DM 99,–
Gehäuse	DM 99,–
Cartridge für SQ555	DM 239,–

Weitere Modelle sowie sonstige Soft- und Hardware auf Anfrage!

CALTEC.
Datensysteme

Eugenstraße 28
7302 Ostfildern 4
Telefon 0711/4579623
Telefax 0711/4569566

Vergleichstabelle der vier Resource Construction Sets

	RCS 1.x	RCS 2.x	K-Resource	Wercs
Auflösung	L/M/H	L/M/H	M/H	L/M/H
Großbildschirm	ja	ja	ja	ja
fehlende Objekttypen	ProgDef Free Strings Free Images extended Bits extended Obtype	ProgDef extended Bits	ProgDef	extended Bits
Wertung	5	7	10	8
Kompatibilität	entfällt	aufwärts	zu RCS x.x	aufwärts
Definitionsdatei	-	5	6	8
Sprachen-Ausgabe	C/Pascal	C/Pascal/ BASIC [2]	C/Pascal	C/Pascal
Definitionsdatei	C-Code [1]	Modula/ BASIC/C-Code [1]	Modula/ Fortran	Modula/ Fortran
Funktions-vielfalt	5	7	8	10
Testfunktion	Nein	Nein	Ja	Ja
Icon-Editor	-	-	4	7
Laden von Icons	Ja[4]	Ja[4]	Nein[4]	Nein[4]
einstellbare Warnstufe	Ja	Ja	Ja	Ja
Beispiele	-	-	8	8
Dokumentation	-	?	46 Seiten 5	95 Seiten 8
Support	Nein[5]	Ja[6]	schriftlich	Ja
Zusatzprogramme	STCreate	STCreate DEF2DFN	-	WIMAGE WCONVERT
Preis	15,-	[3]	129,-	99,-
Bezugsquelle	A	[3]	B	C

[1] C-Code, der die Resource-Struktur enthält, die geändert werden und in Zusammenhang mit STCreate in RSC-Datei umgesetzt werden kann.
[2] in Spezial-Version bei GFA-BASIC mitgeliefert
[3] nicht einzeln erhältlich
[4] mit Icon-Design möglich
[5] über Hot-Line Donnerstags von 14-17 Uhr der ST-Computer-Redaktion
[6] In Zusammenhang mit Support der jeweiligen Programmiersprache

A: MAXON Computer GmbH
Industriestr. 26
6236 Eschborn
Tel.: 06196/481811

B: Gerhard Knupe
GmbH & Co KG
Güntherstr. 75
4600 Dortmund
Tel.: 0231/527531

C: CCD
Burgstr. 9
6228 Eltville
Tel.: 06123/1638

nicht besonders gut zu erkennen. Dennoch wird die Handhabung des Programms zufriedenstellend beschrieben, so daß ein Arbeiten nach dem Lesen der Beschreibung sicherlich möglich ist, zumal das K-Resource 2 auch relativ leicht zu bedienen ist. Gut finde ich auch, daß das Format der Definitionsdatei im Anhang vorhanden ist, wodurch eine Anpassung an andere Formate letztendlich möglich wird. Wenn auch die Einleitung zurecht feststellt, daß man 'in diesem Handbuch keine Programmieranleitung für das AES geben könne', so ist meine Meinung, daß auf Objektstrukturen viel zu wenig eingegangen wird (praktisch gar nicht). Das einfache Mitliefern eines kurzen Beispielprogramms bei einem Programm, das immerhin 129,- DM kostet, finde ich etwas wenig. Wer allerdings sowieso die (hinten im Anhang aufgeführte) Literatur besitzt, sollte sich davon nicht abhalten lassen.

Schon besser hat mir die Anleitung von Wercs gefallen, die sich beispielhaft durch solche Tabellen wie die in Bild 9 auszeichnet (veröffentlicht mit freundlicher Genehmigung von CCD). Außerdem gibt sie meines Erachtens eine insofern ausreichende Einführung in Objektstrukturen, daß wenigstens ein einigermaßen geübter Programmierer sich darunter etwas vorstellen kann. Die Beschreibung des Programms ist im allgemeinen gut (nur die zusammengehörenden Funktionen Einfügen und Importieren sollten in Zukunft etwas ausführlicher dargestellt werden). Auch die im Anhang befindliche Auflistung der Tastaturbefehle sowie des HRD-Dateiformats sind ganz nützlich. Inhaltsverzeichnis und Index sind zufriedenstellend, wobei mir ein Index kaum zu lang sein kann.

Resümee

Nachdem ich nun mehrere Wochen mit allen vier Resource Construction Sets gearbeitet habe und ich mich nun entscheiden müßte, welches ich mir kaufen würde, könnte ich keine einfache Antwort geben (ich weiß, ich mag auch keine Tests, in denen man keine echte Entscheidungshilfe bekommt). Zunächst einmal fällt das RCS 2.x als Kauf weg, da es nur in Zusammenhang mit Entwicklungssystemen erhältlich ist. Bleiben noch das ATARI RCS 1.x, erhältlich als Sonderdisk bei MAXON, das K-Resource 2, zu erwerben bei Knupe, und das Wercs von Hi-Soft, welches in Deutschland von CCD vertrieben wird.

Das ATARI RCS 1.x ist für sein Alter und seinen Preis sicherlich noch gut in Schuß. Wenn es auch nicht gerade jeden Komfort bietet, so hat man (inklusive eines Icon-Editors für 15 DM) für 30 DM sicherlich ein System, mit dem man ohne weiteres arbeiten kann (ich habe jahrelang ohne Probleme damit gearbeitet). Das Preis/Leistungsverhältnis ist recht gut. Nachteilig ist, daß es erstens nicht mehr weiterentwickelt wird (die Ablösung 2.x gibt es wie gesagt nicht einzeln zu kaufen) und zweitens (außer der Sonderheft-Anleitung) keine Anleitung vorhanden ist.

Das K-Resource 2 ist von der Mausbedienung sehr komfortabel und auch die Einführung der objektanhängigen Pull-up-Menüs ist angenehm. Dennoch möchte ich als negative Punkte die etwas schlechtere Anleitung, den meines Erachtens überarbeitungsbedürftigen Icon-Editor und nicht zuletzt den höchsten Preis innerhalb der getesteten Resource Construction Sets anführen.

Das Wercs hat mich ein wenig überrascht, da es (jedenfalls mir) relativ unbekannt war und dennoch ganz gut mit dem

K-Resource 2 mithält. Die Bedienung des Wercs ist nicht ganz so (Maus- und Fenster-)komfortabel wie die des K-Resource 2, was aber durch die umfangreiche Tastenbedienung wieder etwas ausgeglichen wird. Mit seinem Preis von 99,- DM ist es sicherlich eine Alternative zum K-Resource 2, wobei man abwägen muß, ob man eher ein Tastenbedienung mit vielen Zusatzfunktionen wie in Wercs haben möchte, wobei nach Auskunft von CCD voraussichtlich das Zentrieren von Objekten, wie es in RCS 2.x zu finden ist, implementiert wird (nachfragen!), oder einem das Fensterln und Mausen mehr zusagt, und man Wert auf die 'extended' Bits legt.

Alles in allem gibt es inzwischen vier Resource Construction Sets, mit denen man recht gut arbeiten kann. Zur leichteren Kaufentscheidung habe ich Ihnen nochmal eine Tabelle zusammengestellt, in der ich versucht habe, einzelne Punkte innerhalb einer Skala von 1 bis 10 zu bewerten. Natürlich ist eine solche Tabelle subjektiv. Wenn Sie aber abwägen, wie wichtig oder unwichtig Ihnen ein Tabelleneintrag ist, kann diese Tabelle sicherlich trotzdem eine Hilfe sein.

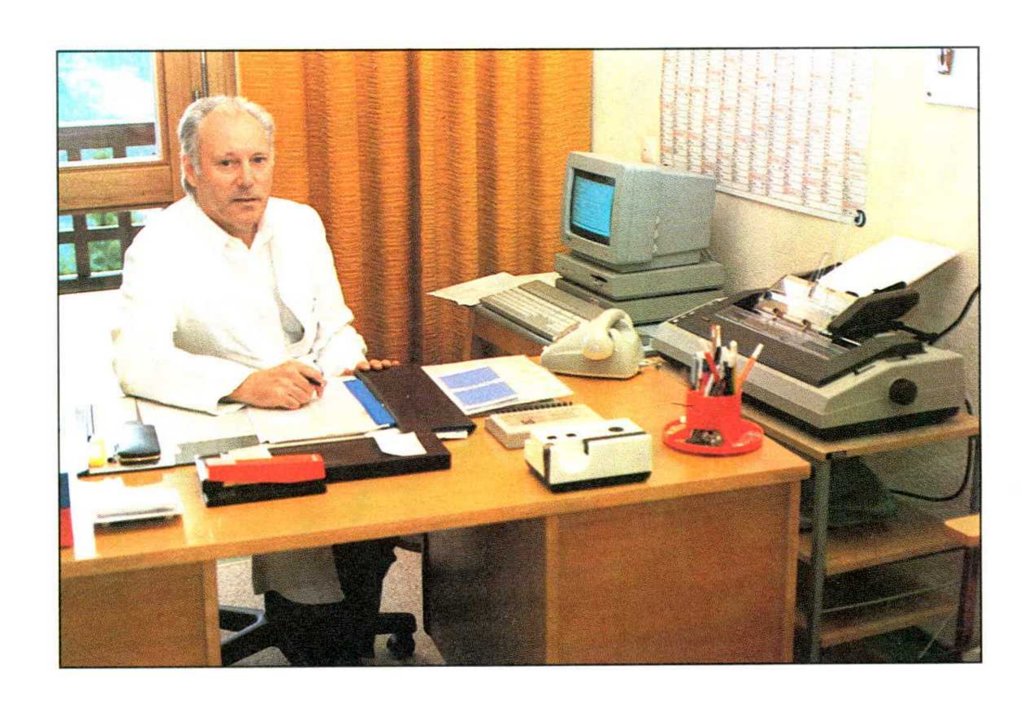
SH

	Default	Exit	Editierbar	Crossed	Disabled	Schattiert
Box	☞	☞	💣	✓	✓	✓
BoxChar	☞	☞	💣	✓	✓	✓
BoxText	☞	☞	💣	✓	✓	✓
Button	✓	✓	💣	✓	✓	✓
FBoxText	☞	☞	☞	✓	✓	✓
FText	☞	☞	☞	✓	✓	✗
IBox	☞	☞	💣	✗	☞	✓
Icon	☞	☞	💣	✓	✓	✗
Image	☞	☞	💣	✓	✓	✗
ProgDef	☞	☞	💣	✓	✓	✗
String	☞	☞	💣	✓	✓	✗
Text	☞	☞	💣	✓	✓	✗
Title	☞	☞	💣	✓	✓	✗

Schlüssel:

- ✓ Ändert Darstellung des Objektes und Verhalten des AES.
- ☞ Ändert Verhalten des AES aber nicht die Darstellung.
- ✗ Hat keine Auswirkungen.
- 💣 Bewirkt einen Absturz des Rechners (Bomben).

Bild 9: Eine Tabelle aus der Wercs-Anleitung stellt den Zusammenhang zwischen Objekttyp und Auswirkung dar.



Assoziatix

oder mehr Zeit für Medizin

Südseitig auf 1400 m Seehöhe liegt in einer der sonnenreichsten Gegenden Europas das Krankenhaus Stolzalpe. Wer könnte vermuten, wenn er dieses markante Bauwerk erblickt, daß sich dahinter eines der führenden orthopädischen Zentren Österreichs verbirgt? Daß der ATARI ST in diesem Hause für medizinische Anwendungen akzeptiert wird, hat schon der ärztliche Direktor Professor Dr. Reinhard Graf bewiesen. Er erfand und entwickelte die Hüftsonographie beim Säugling. Diese ungefährliche Vorsorgeuntersuchung dient dazu, angeborene Hüftleiden frühzeitig zu entdecken, die nur in dieser Phase optimal behandelt werden können. Bei dieser wichtigen Ultraschalluntersuchung ist nur ein einziges Schnittbild wirklich zur Auswertung geeignet. Und genau hier wird der ATARI ST zum Speichern und Analysieren der Einzelbilder eingesetzt. Doch damit nicht genug. Seit 1988 hat der 'ST' durch die Initiative von Prim. Dr. Béla Farkas ein neues Anwendungsgebiet gefunden. Die Entwicklungszeit betrug knapp 3 Jahre, das Ergebnis heißt 'Assoziatix' und ist mittlerweile als Sonderdisk bei MAXON erhältlich (siehe Sonderdisk-Seiten in dieser Ausgabe).

Notarzt: Datenmenge kontra Geschwindigkeit

Die grundlegende Idee zum Projekt Assoziatix geht zurück auf den Herbst 1986. Damals standen wir, eine Gruppe von jungen Ärzten, vor dem Problem, die Einsatzprotokolle der Notarztwagen des ÖRK in Graz mit einem erschwinglichen Computersystem zu verwalten und gezielt auszuwerten.

Als geeignete Hardware erwies sich recht bald der ATARI 1040 ST. Größere Schwierigkeiten gab es allerdings bei der Wahl der Software, da wir einerseits doch einige Anforderungen stellten, und andererseits das Angebot an Datenverwaltungen damals recht dünn gesät war. Unsere Datenmenge bestand, grob geschätzt, aus 3000 - 4000 Datensätzen, wobei pro Satz ein Karteiblatt und ein Notarztprotokoll geführt werden mußten. Dieser Einsatzbericht in Form eines Fragebogens sollte mindestens 400 - 500 'ankreuzbare' Punkte umfassen können. Die Suche in den Protokollen sollte das Finden von bestimmten Mustern zulassen, wobei verschiedene logische Verknüpfungen möglich sein mußten. Die Geschwindigkeit, mit der die Datenmenge (1,5 Mio. Punkte) durchforstet werden sollte, mußte erträglich sein (möglichst nur einige Sekunden). Das war unseres Wissens mit einer konventionellen, dialog-orientierten oder programmierbaren Datenbank nicht zu erreichen. Nach einem Jahr Entwicklung und Probelauf konnten wir beginnen, die Kinderkrankheiten des Programmes zu 'behandeln'. Gleichzeitig fand sich ein neues Einsatzgebiet für Assoziatix.

Der Einzug ins Krankenhaus

Das Institut für Anästhesiologie auf der Stolzalpe unter Leitung von Primarius Dr. Béla Farkas muß jährlich rund zwei- bis dreitausend Operationen betreuen. Bei dieser Tätigkeit fallen neben den Stammdaten des Patienten auch medizinische Daten an. Diese wurden nach einem Diagnoseschlüssel codiert und archiviert. Mit 'Lochkarten' und Selektierer war eine Bearbeitung möglich, allerdings alles andere als benutzerfreundlich. Für jede Fragestellung mußten die Lochkarten gezielt auf einen Punkt hin untersucht werden. Bei einer neuen Frage mußte unter Umständen der gesamte Aufwand

wiederholt werden. Daher suchte Dr. Farkas nach einem anderen Dokumentationssystem. Assoziatix bot sich ideal zur Verwaltung der Daten, die als Mischung aus Texteingaben und Ja/Nein-Abfragen vorlagen, an. Alle medizinisch relevanten Informationen über die Narkose, wie z. B. Narkoserisiko oder prä- und postoperative Komplikationen werden nun über ein standardisiertes Narkoseformblatt erfaßt. Dieses Formblatt, einem Fragebogen ähnlich, wird durch Ankreuzen der zutreffenden Punkte für jeden Patienten angelegt.

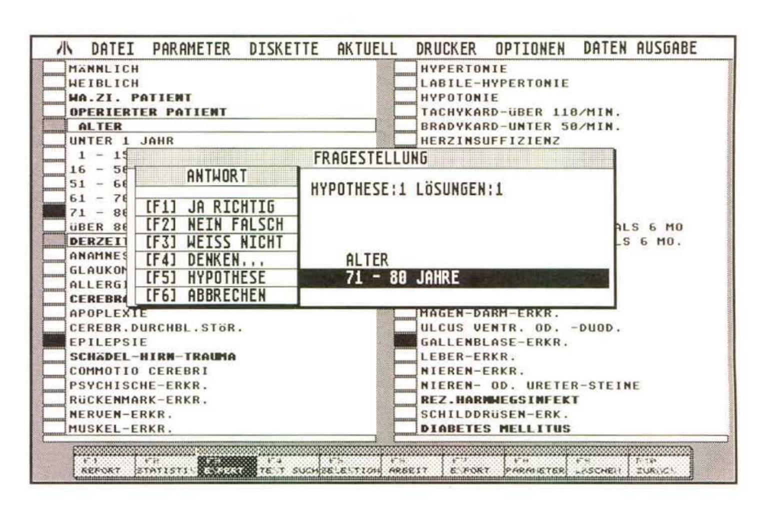


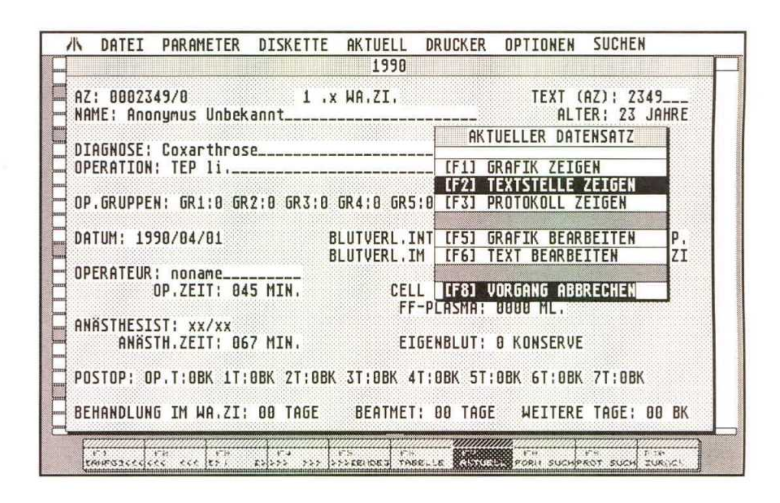
Das Krankenhaus Stolzalpe

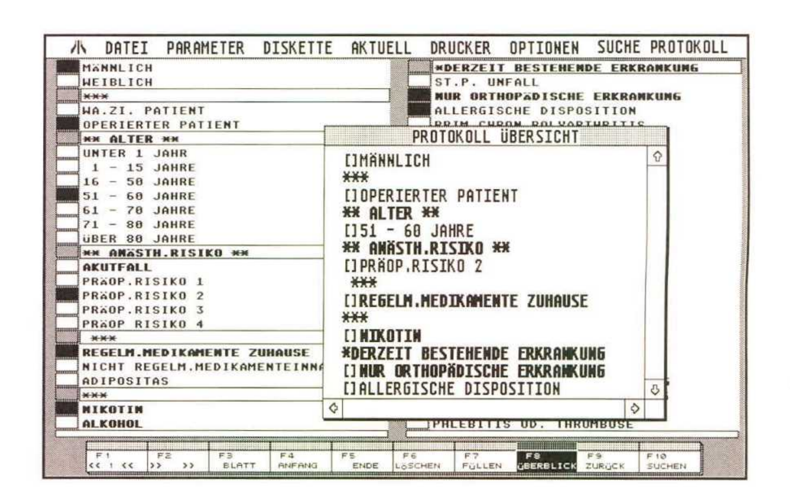
Ideen gegen Realitäten

Nach intensiven Diskussionen mit Prim. Farkas waren seine Vorstellungen und meine Möglichkeiten definiert. Nun galt es, die Ideen, die mit der Zeit immer gewagter wurden und schon begannen, die Hardware zu sprengen, wieder an die Hardware - einen 'Single User'- und 'Single Tasking'-Rechner - und an meine Programmiertechnik zurückzuführen. Jede Möglichkeit der Verbesserung und Anpassung rief eine neue Idee hervor, nahezu eine Sisyphusarbeit. Dieser Aufwand wurde aber dadurch entschädigt, daß Assoziatix ständig als Testversion lief und ein laufendes Feedback über auftretende Probleme gegeben war.

Als erster Punkt mußte die Datensicherheit erhöht werden. Jeder Datensatz wird daher sofort auf die Festplatte ausgelagert. Ein Absturz oder Stromausfall sollte vom Programm erkannnt und behandelt werden. Dies wurde so gelöst, daß Assoziatix, sofern eine Datei nicht ordnungsgemäß abgeschlossen wurde, selbständig eine Restaurierungsmöglichkeit beim Neustart anbietet. Vor und während der Operation werden die Daten auf einem Anästhesieprotokoll schriftlich erfaßt. Der Aufbau dieses Erhebungsblattes entspricht dem Bildschirm-Layout, damit die Übertragung in den Computer auch für den Ungeübten keine Schwierigkeiten macht. Zur Vervollständigung wurde von Dr. Farkas auch ein Eingabehandbuch aufgelegt. Das Verständnis für den Computer bei den Mitarbeitern war so problemlos gegeben.







Das Resultat

Die Speicherung der Protokolldaten erfolgt für jede Ja/Nein-Abfrage als Bit. Diese Form bringt zwei entscheidende Vorteile: zum einen komprimierte Speicherung des Protokolls mit Minimierung des Speicherbedarfes, zum anderen wird auf diese Weise eine schnelle Suche ermöglicht, indem 32 Bits zu einem Langwort zusammengefaßt werden. Da Lang-

wortoperationen auf den 68000-Prozessor optimal zugeschnitten sind, werden bei der Suche nur noch Langwörter miteinander verglichen.

Assoziatix kann nicht nur exakte Begriffe in Datenfeldern suchen,sondern durchforstet den Datenbestand auch nach Merkmalen, die einen Datensatz beschreiben. Diese ergeben als Vorgabe ein Muster. Da jedes beliebige Muster gesucht werden kann, muß die Suche im Protokoll sequentiell durchgeführt werden. Und genau diese Suche erfordert bei relationalen Datenbanken viel Zeit und wurde daher in Assoziatix optimiert. Die Protokollsuche erlaubt auch logische Verknüpfungen, wie z.B. "finde alle Patienten: 'männlich' und ('1-15 Jahre' oder 'präoperatives Risiko 4')".

Als zusätzliches Extra können sogar ähnliche Muster gesucht werden, die sich z.B. in einer definierbaren Anzahl von Punkten von der Vorgabe unterscheiden dürfen.

Hin und wieder gibt es Zusatzinformationen zu einem Patienten, die nicht in einem Datensatz untergebracht werden können. Daher der Wunsch von Dr. Farkas, auch Texte und eventuell kleine Skizzen mitzuverwalten. Ich wußte freilich nicht, welche Mühen und welches Kopfzerbrechen mit dieser Vorgabe gekoppelt sein sollten. Aber mein Ehrgeiz war geweckt und die Idee zu einem in die Benutzeroberfläche integrierten Texteditor und Grafikprogramm wurde realisiert. Als goßes Plus kann nun jederzeit aus einem Datensatz heraus ein dazugehöriger Text und/oder eine Grafik auf 'Knopfdruck' bearbeitet werden. Die lästigen Wartezeiten für das Nachladen eines Editors entfallen.

Ein weiteres 'Zuckerl', welches auf Anregung von Dr. Farkas - zu meinem Leidwesen als Programmierer - implementiert wurde, ist die 'Experten-Funktion'. Es kommt immer wieder vor, daß nur vage Erinnerungen an einen Patienten existieren. In diesem Fall kann man sich vom Computer nach Merkmalen zum Datenbestand befragen lassen. Aufgrund der Antworten des Anwenders werden die nicht zutreffenden Hypothesen eliminiert, bis im Idealfall nur mehr der gewünschte Datensatz übrigbleibt.

Im Routinebetrieb ging bei komplexen, schrittweisen Suchanfragen öfters die Fragestellung verloren. Wir hatten zwar ein Ergebnis, konnten aber die einzelnen Abfrageschritte nicht mehr 'aus dem Kopf' nachvollziehen. Auch dieses Problem wurde gelöst. Nun werden diese Tätigkeiten automatisch im Klartext aufgezeichnet und können jederzeit problemlos im Texteditor überprüft und manipuliert werden.

Endlich fertig?

Seit Sommer 1988 läuft Assoziatix im praktischen Einsatz im Krankenhaus Stolzalpe. Durch kontinuierliches Feedback zwischen Anwender und Programmierer konnten laufend wertvolle Erfahrungen in die Weiterentwicklung einfließen. Heute werden die unzähligen schlaflosen Nächte durch das Wissen, das aufgrund des konsequenten EDV-Einsatzes aus den Rohdaten gewonnen werden kann, mehr als wettgemacht. Anästhesiologische und intensivmedizinische Daten können nun jederzeit für wissenschaftliche Zwecke abgerufen werden. Die jährlich vom Spitalserhalter geforderte Statistik wurde 1989 bereits vollständig mit Assoziatix erstellt. Der dafür erforderliche Zeitaufwand schrumpfte von einigen Wochen auf einige Stunden.

Der Routineeinsatz klappt nun bestens, aber die Geister die man einmal rief, die wird man nicht mehr los. Dr. Farkas hat bereits neue Ideen...

Dr. Wolfgang Kohlmaier

Hotwire

Der Programmstarter



Wenn der ST intensiv genutzt wird, steht der Anschaffung einer Festplatte nichts mehr im Wege. Doch damit beginnen auch die ersten Probleme: "Wo war denn noch das Programm?" oder "Welchen Parameter muß ich jetzt noch übergeben?" sind häufig gestellte Fragen, mit denen man sich als Anwender herumschlagen muß. Hotwire verspricht, auf Tastendruck aus dem Desktop Programme starten zu können, um der ewigen Sucherei ein Ende zu bereiten. Lesen Sie, ob und wie es funktioniert.

Die ewige Suche nach den Programmen und den dazugehörigen Pfaden soll nun endlich ein Ende haben - Hotwire macht's möglich. 1st_Wordplus? Natürlich, Control-Alt-W. Stad? Wo war es denn noch? Na, egal, nur kurz Alternate-S drücken. Diese Sätze müssen keineswegs ein Wunschtraum bleiben, sondern sind bereits Alltag bei Hotwire-Anwendern. Interessant dabei ist, daß Mitteilungen wie "Der Druckertreiber kann nicht gefunden werden" (Wordplus) nicht mehr auftauchen, denn bevor ein Programm gestartet wird, wechselt Hotwire auf das entsprechende Laufwerk und wählt den korrekten Pfad an. Auf diese Weise können alle Programme ihre Resource-Dateien, Druckertreiber usw. problemlos finden.

Freie Auswahl

Hotwire kann, wie bereits gesagt, Tastenmakros belegen. Wenn man eine Kombination nicht mehr weiß, genügt ein Druck auf die rechte Maustaste oder Shift-Help, um alle möglichen Kombinationen anzuzeigen. Hier zeigen sich auch die verschiedenen Möglichkeiten, die durch Hotwire angeboten werden. Um ein Programm wiederfinden zu können, kann ein beliebiger Name von bis zu 20 Zeichen Länge gewählt werden. Auf diese Weise ist das Aufspüren von Programmen bereits wesentlich einfacher als im Desktop, in dem bekanntlich nur 8 Zeichen plus drei für die Extension zur Verfügung stehen. Zu installierende Programme können komfortabel mit der Fileselectbox ausgewählt werden, lästige Pfadeingaben per Hand entfallen also. Nachdem der Titel aus der Auswahlbox ausgewählt ist, kann man ein (fast) beliebiges Tastaturkürzel dafür reservieren. Um Doppelbelegungen zu vermeiden, mäkelt Hotwire diese sofort an. Wählbar sind alle Kombinationen aus Shift links, Shift rechts, Control, Alternate und einem Buchstaben oder einer Taste. Das Kürzel kann entweder per Maus festgelegt oder direkt über die Tastatur eingegeben werden. Sollte das Kürzel Ihnen entfallen sein, können Sie durch einen kurzen Druck auf die rechte Maustaste nachsehen, wie ein bestimmtes Programm zu starten ist: Hotwire zeigt in seinem Auswahlmenü alle installierten Programme inklusive Tastaturkürzeln an.

GEM-, TOS- und TTP-Programme können durch einen jeweiligen Button festgelegt werden. Diese Einstellung ist wichtig, um einen bei TOS-Programmen erscheinenden Cursor nach Beendigung des Programms wieder auszuschalten - ein weiterer Pluspunkt für Hotwire. Kennen Sie auch das Problem, daß nach dem Aufruf eines TOS- oder TTP-Programms aus dem Desktop der Bildschirm so schnell gelöscht wird, daß man oft die Dinge nicht mehr lesen kann, die man gerne hätte lesen wollen? Hier zeigt Hotwire einen weiteren Pluspunkt: Wird der Button "Rückfrage" aktiviert, bleibt der Bildschirm solange erhalten, bis eine beliebige Taste gedrückt wird. Werden Programme aufgerufen, die eine Programmzeile benötigen, kann man auch das angeben. Vor dem Start des jeweiligen Programms muß man dann noch die gewünschte Parameterzeile eingeben, die übrigens länger ist als die Desktop-eigene Eingabezeile: 64 Zeichen. Hat man ein Programm, dem immer die gleichen Parameter übergeben werden sollen, kann man diese auch fest speichern. Bis zu zehn feste Parameterzeilen können dauerhaft abgespeichert werden und stehen damit rund

um die Uhr zur Verfügung.

Normalerweise lassen sich über das Desktop Anwendungen anmelden, so daß beispielsweise bei einem Doppelklick auf eine Datei mit der Endung "DOC" 1st_Wordplus nachgeladen wird. Hotwire erlaubt, bis zu vier solcher Endungen für ein Programm zu installieren. Dadurch wird die Vielfalt der anklickbaren Dateien größer, bei unserem Beispiel könnten also auch Dateien mit der Endung "BAK" für Wordplus installiert werden.

HotWire! © 1989 CodeHead Software

Stad	^A	ARC	^A	--------------------
BTX-Manager	^B	LHarc	^L	--------------------
MultiAcc	^C	ZOO	^Y	--------------------
Flash	^F	RZ	^R	--------------------
Menu X	^G	SZ	^S	--------------------
Hotwire	^H	Rufus	^U	--------------------
Fileselectbox	^I	--------------------		--------------------
Harlekin-Konfig	^K	--------------------		--------------------
Look	^L	--------------------		--------------------
MagicBox ST	^M	--------------------		--------------------
Interlink	^N	--------------------		--------------------
Oma's Basic	^O	--------------------		--------------------
Script	^P	--------------------		--------------------
TurboST 1.8	^R	--------------------		--------------------
SED 4.40	^S	--------------------		--------------------
Tempuz	^T	--------------------		--------------------
Unitern	^U	--------------------		--------------------
1st_Wordplus	^W	--------------------		--------------------

LOAD SAVE MULTI-DESK ENDE

Bild 1: Das Auswahlmenü von Hotwire. Deutlich sind die Tastaturkürzel zu erkennen.

Pipeline

Zwischen Hotwire und Multidesk (ein Programm, das ebenfalls über Bela vertrieben wird) existiert eine Pipeline. So gibt es in Hotwire einen Button, der Multidesk aufruft, sobald er angeklickt wird. Dadurch wird der lästige Weg bis zur linken Menüleiste erspart. Hotwire registriert automatisch, ob Multidesk installiert ist oder nicht. Das Zusammenspiel zwischen den beiden Programmen sollte ein Beispiel für zukünftige Programme sein.

Sofern Sie noch kein neues TOS besitzen, aber trotzdem ein GEM-Programm nach dem Booten automatisch starten wollen, stellt Ihnen Hotwire auch dafür eine komfortable Möglichkeit zur Verfügung, denn eines der installierten Programme kann als Autostart-Programm installiert werden und wird dadurch direkt nach dem Start des Rechners automatisch ausgeführt. Dabei ist es nicht von Bedeutung, ob das Programm ein TOS-, TTP-, APP- oder GEM-Programm ist.

Programmverkettungen sind ebenfalls möglich. Angenommen, Sie haben eine Reihe von Programmen, die automatisch nacheinander gestartet werden sollen. Als erstes Programm benutzen Sie eine Textverarbeitung, um einen Text zu schreiben. Das nächste Programm entfernt harte Returns aus dem soeben geschriebenen Text, während sich das dritte und letzte Programm darum kümmert, den verfaßten Text in einen Fließtext umzuwandeln. Mit Hotwire läßt sich bei jedem Programm ein Verweis auf ein anderes anbringen, das nach Beendigung desselben gestartet werden soll.

Nachladbar

Durch die Möglichkeit, Konfigurationsdateien nachzuladen, läßt sich die gesamte Konfiguration von Hotwire durch wenige Tastendrücke ändern. Dadurch können beispielsweise mehrere Benutzer Hotwire anwenden, ohne andauernd neue Tastaturmakros erstellen zu müssen. Ebenso einfach können verschiedene Dateien für unterschiedliche Anwendungsbereiche erstellt werden. So lassen sich in der einen Datei DTP-Programme unterbringen, in der zweiten Grafikprogramme, in der dritten Spiele usw. Dadurch wird Hotwire fast universell einsetzbar.

Der "Affengriff" zum Reset-Taster dürfte mittlerweile allen Mega ST-Benutzern bekannt sein. Für die geplagten Mega-Anwender, die (noch) kein TOS 1.4 besitzen, bietet Hotwire die bekannten Tastenkombinationen Control-Alt-Delete für den Warm- und Control-Alt-Shift-Delete für den Kaltstart. Hotwire-Anwender haben dadurch keine Rückenschmerzen mehr.

Bilanz

Hotwire kostet DM 79,-. Der Preis ist für die Leistungen des Programms nicht zu hoch gegriffen. Hotwire sollte allerdings nur dann angewendet werden, wenn viele Programme auf einem Rechner benutzt werden. Probleme mit anderen Programmen konnten nicht festgestellt werden, auch mit der neuen TOS-Version läuft Hotwire problemlos. Andere Programme, die ebenfalls Tastaturmakros belegen (etwa Harlekin), kommen nicht mit Hotwire ins Gehege und laufen problemlos nebeneinander. Einziges Manko: Die Anleitung verdient nicht das Wort "Anleitung". Hotwire ist jedoch recht leicht zu durchschauen, wodurch die Beschreibung nicht zur wichtigsten Unterlage wird. Alles in allem also ein recht gelungenes Programm.

MP

Bezugsquelle:

Bela Computer
Unterortstraße 23-25
6236 Eschborn
Tel. (06196) 481944

Moderne Zeiten

Mit Tempus Word ins neue Zeitalter der Textverarbeitung



Sie haben das schon tausendmal gelesen: 'Wollte man alle Funktionen des Programms aufzählen, dann... das ganze Heft...' Aber im Ernst: So einfach ist das mit Tempus Word wirklich nicht. Und trotzdem wagen wir mit dem folgenden Beitrag den Schritt in die Ungerechtigkeit.

Vor einiger Zeit beglückte CCD seine Kunden mit einer verheißungsvollen Botschaft. Tempus Word, so hieß es darin, sei nun verfügbar. Zwar eine 0.9-Version, in der noch einiges fehle. Aber immerhin: Der Preis betrage nur 450 DM, die Version 1.0 würde 200 Märker mehr kosten. Entscheiden sollte man sich zügig, denn nur 500 0.9-Exemplare sollten ausgegeben werden.

Und nun ist es tatsächlich soweit: Mittlerweile befindet sich die 0.91-Version von Tempus Word auf meiner Platte, und ich habe die ehrenvolle Aufgabe, Ihnen dieses Programm näherzubringen. Eigentlich müßten Sie sich jetzt wundern. Denn es widerspricht den Gepflogenheiten dieser Zeitschrift, Beta-Versionen zu besprechen. Wir haben uns trotzdem dazu entschlossen. Denn erstens warten sehr viele Leute auf dieses Programm, und zweitens ist das Produkt in dieser Form ja wirklich käuflich. Bevor wir uns aber dem Neuling zuwenden, noch einige Bemerkungen zum Thema des Tages: Textverarbeitung.

Habe ich's mir doch gedacht! Wir beginnen sofort, uns zu streiten. Denn mit 100%iger Sicherheit driften unsere Vorstellungen von der idealen Textverarbeitung völlig auseinander. Und Sie da hinten in der vorletzten Reihe: Was, nur Formulare soll Ihr Traumprogramm beherrschen, ob das auch geht?

Heiße Wünsche

In kaum einem Bereich der Computerei werden solche erbitterten Grabenkämpfe ausgetragen wie bei dem ewigen Streit um die richtige Textverarbeitung. Das kommt ja auch nicht von ungefähr. Während eine Datenbank tatsächlich nur dazu benutzt werden kann, Daten zu sammeln, wird ein Textprogramm viel flexibler eingesetzt, muß sich in vielen verschiedenen Anwendungen bewähren. Noch eins: Den Umgang mit Datenbank und Tabellenkalkulation lernten wir erst mit dem Rechner kennen. Unsere Arbeitsweise ist geprägt vom rechnerorientierten Vorgehen. Doch Texte verfaßt und mit der Schreibmaschine zuwege gebracht, das haben wir schon in den grauen Tagen der computerfreien Vergangenheit. Will sagen: Hier wie sonst nirgends hat sich die Software unseren Vorgaben anzupassen. Und daran scheitern die meisten Programme.

Ich brauche keine Namen zu nennen. Bei manch selbsternannter 'Software' scheitert's doch schon an der Trennung. Also sprach Zarathustra: Was soll's dann? Wir meinten doch immer, daß die Software für uns denkt. Zumindest so weit, wie sie es mit ihren Algorithmen kann. Genug der Polemik.

Da wir also unsere Wünsche nicht artikulieren können, weil's zuviele sind, werfen wir einfach mal alles in einen Topf, was man so bekommen kann im Bereich Textverarbeitung. (Diesen Begriff muß man sich mal auf der Zunge zergehen lassen: 'Text-Verarbeitung'. Also meine Phantasie läßt mich mit Texten vieles anstellen. Doch hübsch der Reihe nach.) Ich teile die Vorgänge in einzelne Häppchen.

Erfassen

Bevor's überhaupt was zu verarbeiten gibt, muß der Text getippt werden. Da führt nun mal nichts dran vorbei. Denn mit Spracherkennung soll es auf dem ST noch Probleme geben. Der Geübte schreibt mit bis zu 10 Fingern und ist in der Lage, sich Tastaturbefehle zu merken. Das Springen auf Zeilenenden, Absätze etc, Setzen von Blockmarkierungen usw. - alles das muß per Tastatur geschehen können. Bei diesen Vorgängen sind Proportionalzeichensätze eher ein Hindernis, verwirren sie doch nur das Positionieren der Schreibmarke (landläufig: Cursor).

Die Gedanken sind meist schneller als die Finger, Textbausteine sollen per Blockbefehle schnell gegeneinander verschoben werden können. Tastaturmakros, die auf Tastendruck z.B. die Anrede einspielen - sie helfen, dem Elaborat die klare gedankliche Linie zu verleihen, ohne wie früher alles noch einmal zu tippen. Eine solche Hilfe ist auch das pure Abschreiben (sorry: Kopieren) aus anderen Texten. Mehrere Texte sollten also gleichzeitig bearbeitet werden können.

Gestalten und Verfeinern

Die Schrift auf dem Papier, sie hat ja ein Format: Ränder von der Blattkante, einen bestimmten Font, der das Aussehen bestimmt, einen Zeilenabstand, Formate wie Blocksatz oder Flatterrand. Bestimmte Passagen sollen durch Kursivschrift hervorgehoben, die Überschrift unterstrichen werden. Hier öffnet sich der weite Bereich der Schriftattribute.

Doch oft genug reicht das nicht. Ein Absatz soll eingerückt werden, für die Adresse muß der Text ein wenig über den linken Rand ragen, oder, oder. Spezielle Absatzformate müssen her, für die sämtliche Parameter, die das Aussehen des Textes bestimmen, noch einmal gesondert festgelegt werden können.

Spaltensatz wäre auch nicht schlecht, macht allerdings mehr als sonst noch Trennung erforderlich. Die sollte wiederum ohne Eingriff von außen möglichst fehlerfrei arbeiten. Mehrere Fonts gleichzeitig in einem Dokument zu benutzen - Natürlich alle in Proportionalschrift und das in echtem "wie-man's-sieht-so-wird's" am Bildschirm angezeigt. Das ewige Problem: Das Seitenende trennt einen Absatz. Oder die Überschrift ist hüben, der Text dazu drüben. Immer wieder das lästige Setzen des Seitenendes zu Fuß. Das könnte das Programm in seiner Quasi-Intelligenz eigentlich selbst tun. Man legt nur fest, in welchem Bereich Absätze getrennt werden dürfen, die Entscheidung liegt dann bei der Software.

Illustrieren

Ach ja, Bilder sollten natürlich auch noch hinein. Sie sollen aus jedem Malprogramm zu übernehmen sein, Ausschnitte sollen gesetzt und gezoomt werden können. Aber auch Vektorgrafik gleich einbinden zu können, das wär' nicht übel. Es bringt uns allerdings auch fast schon in den Bereich des DTP. Aber es gibt noch etwas Grafisches: das freie Positionieren von Text à la Signum!, das Formelsatz erst möglich macht. Das heißt: die Zeichen können am Schirm verschoben werden, sind dennoch als solche zu identifizieren.

Drucken

Die Ausgabe ist eigentlich das wichtigste. Denn letztlich zählt das gedruckte Ergebnis und nicht die Arbeit, die man aufwandte, um dahinzukommen. Als Ausgabegerät werden bei heutigen Qualitätsansprüchen fast nur noch 24-Nadel- oder Laserdrucker in Frage kommen. Auf deren Fähigkeiten muß eingegangen werden. Für den Korrekturabzug reicht allemal die druckerinterne Schrift. Doch die finale Qualität erreicht man nur per Grafikdruck. Zumal so auch die verschiedenen Fonts ermöglicht werden. Das hat rückwirkend natürlich Einfluß auf das Gesamtkonzept des Programms. Denn nun müssen Font-Formate gefunden, Anbindungsmöglichkeiten an bereits bestehende geprüft werden etc...

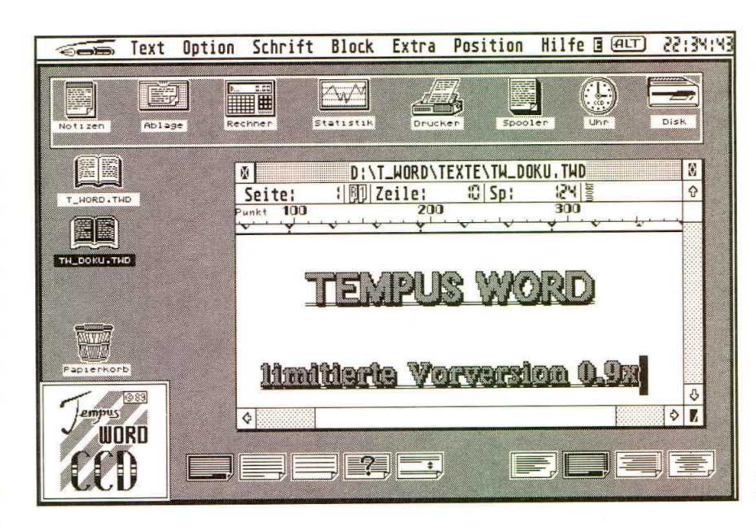


Bild 1: Der Desktop von Tempus Word

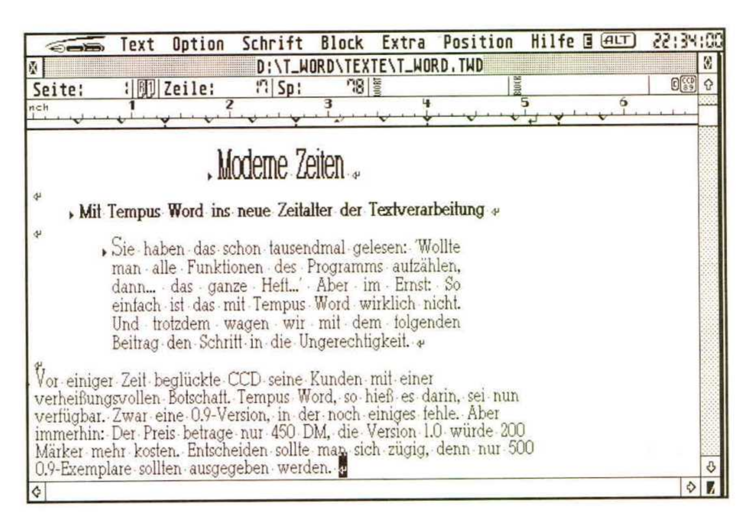


Bild 2: Darstellungen von voller Grafik...

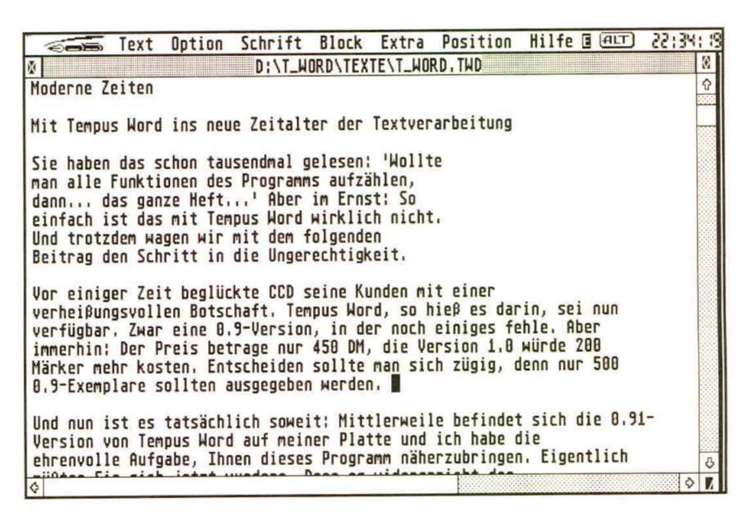


Bild 3: ... bis hin zu kargem Text

Am laufenden Band

Soweit, so gut. Der Text ist gedruckt. Das Programm hat seine Schuldigkeit getan. Oder auch nicht. Sinn und Nutzen der computergestützten Textverarbeitung beginnen oft erst hier: Das Anschreiben hunderter Kunden sollte eine Serienbrieffunktion ermöglichen. Dabei tritt das Problem der Datenübernahme aus anderen Programmen auf. Denn die Adressen sind in einer Datenbank gespeichert.

Formulare

Während beim Serienbrief die Software den Text einsetzt, sind bei Formularen nur bestimmte Stellen überhaupt beschreibbar. Der restliche Text bleibt fest. Beim Rechnen werden Zahlen, die an bestimmten Tabulatorpositionen stehen, über Rechenfunktionen miteinander verknüpft. Anwendung findet die Kombination beider Funktionen beim Schreiben von Rechnungen.

Verwalten

Doch damit nicht genug. Was nützt mir eine Platte mit Megabytes an Texten, wenn ich mich darin nicht zurechtfinde? Das Programm sollte in der Lage sein, die Archivierung der Texte zumindest zu erleichtern. Suchfunktionen auch für nicht geladenen Texte etc. ermöglichen das Aufspüren bestimmter Passagen und verhinden so, daß Dinge nochmal geschrieben werden.

Bedienung

Über alledem steht jedoch die Bedienbarkeit der Funktionen. Alles muß klar, einprägsam und im logischen Kontext serviert werden. Sonst schmecken die Menüs nicht, und all die schönen Möglichkeiten bleiben ungenutzt. Ob nun mit Maus oder Tasten bedient wird, sei dahingestellt. Am besten unterstützt das Programm alle Eingabevarianten. Häufig wiederkehrende Bedienungsabläufe sollten zu Makros zusammengefaßt werden können. Die Makros kann man mit Bedingungen und Sprüngen zu einer Quasi-Programmiersprache ausbauen. Das klingt zwar spinnig. Doch auf diese Weise können Bedienung und Funktion auf sehr spezielle und individuelle Bedürfnisse zugeschnitten werden. Und das macht eine Software erst flexibel.

Intelligente Hilfen

Bis hier sind die Funktionen, die ein Textsystem zur Verfügung stellen sollte, rein formaler Art. Sie beziehen sich nicht auf den Inhalt des Textes. Dergleichen ist ja den Rechnern auch ungleich schwerer beizubringen. Doch welcher Rechtschreibschwache wünscht sich nicht eine automatische Fehlererkennung, vielleicht sogar schon beim Schreiben? Sie erfordert ein Lexikon, gerade für die deutsche Sprache eine nicht triviale Aufgabe für Progammierer. Ein Thesaurus sucht nach Worten ähnlicher oder konträrer Bedeutung. Nicht nur für Leute mit geringem Wortschatz geeignet. Lassen sich Bemerkungen in den Text schreiben, die nicht gedruckt werden? Das ist eine Hilfe für alle, deren Texte am Rechner korrigiert werden.

Wiederum formaler Art sind Funktionen wie Fuß- und Endnoten. Der Rechner kann hier - sofern ausreichend zu parametrisieren - eine Menge Arbeit abnehmen. Zu diesem Bereich gehören ebenso Indexlisten und Inhaltsverzeichnisse.

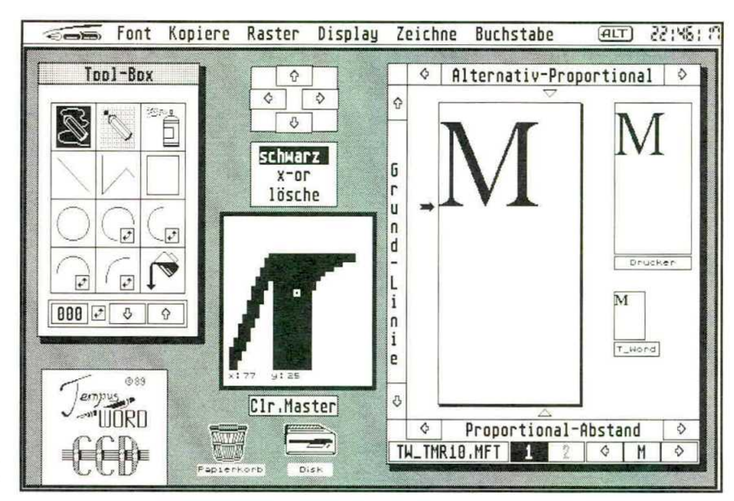


Bild 4: Der Zeichensatzeditor von Tempus Word

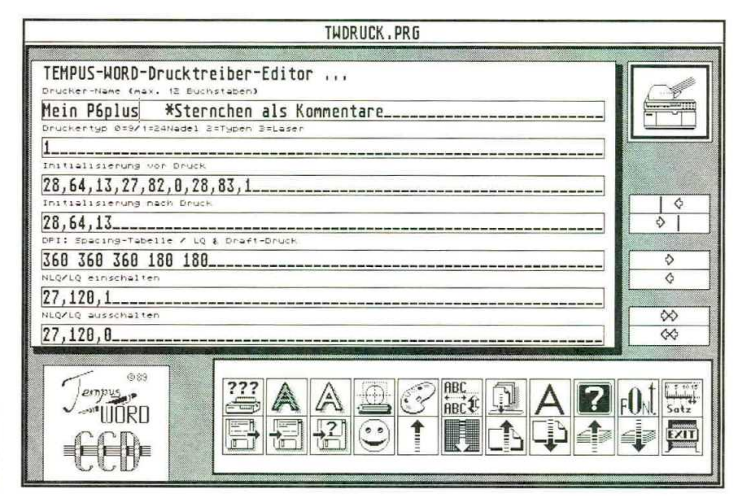


Bild 5: Hier werden Druckertreiber erstellt

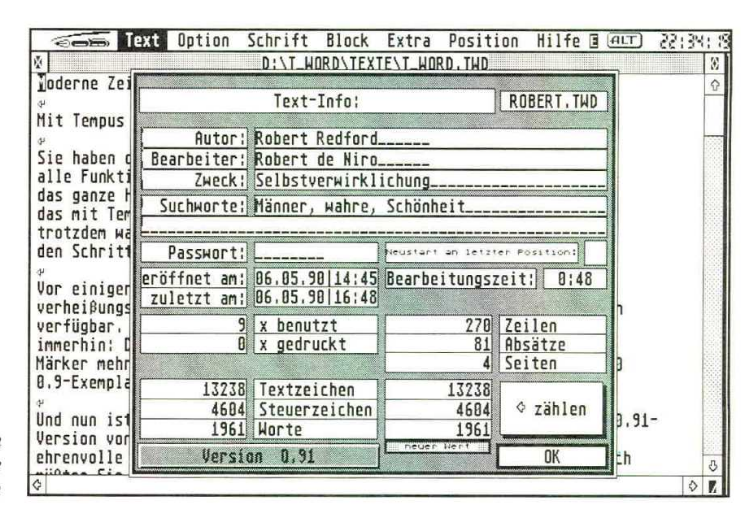


Bild 6: Zu jedem Text die volle Information

In medias res

Soweit allgemein zum Thema Textverarbeitung. Wahrscheinlich haben Sie im Kopf eine Checkliste gemacht und angekreuzt, wieviel davon Sie benötigen und wieviel davon Ihr momentanes Textprogramm beherrscht. Die Differenz dieser Kreuzmengen ist direkt proportional zu Ihrer Unzufriedenheit. Oder? Vielleicht hilft Tempus Word Ihnen. Ich werde also im folgenden nicht das herausstellen, was

FREIHEIT IN D.T.P.



Publishing Partner MASTER

Freiheit der Ausgabe:	Matrix-/Laserdrucker/Postscript/Linotronic
Freiheit der Farbe:	Freie Farbwahl, Ausdruck mit Farbdruck oder Farbseparation, mechanische Separation
Freie Bedienung:	Interaktiv, kein Editor nötig
Freier Textimport:	Calamus CTX, That's Write, WordPerfect, Wordplus
Freier Grafikimport:	NeoChrome, Degas, IFF, Metafile, Tiny, IMG, Arabesque, STAD, AegisDraw Plus, EPSF (Encapsulated PostScript)
Freiheit der Sprache:	Deutsche, Englische, Französische, Spanische Silbentrennung
Freiheit der Maße:	Zentimeter, Millimeter, Zoll, Picas, U.S. Points, Cicero, Didot, Metric Points

Publishing Partner MASTER geht über ein normales DTP Programm hinaus:
Feste und einstellbare Vergrößerung, variabler Zoom, linke/rechte Grundseiten, Ausschneiden und Einfügen auch zwischen verschiedenen Texten, 3-dimensionales Rotieren von Text, Grafiken und Objekten, automatischer und manueller Textfluß, Suchen/Ersetzen (inkl. Attribute), Umfließen von unregelmäßigen Objekten, Stilvorlagen (Layouts, Stylesheet, ...),
...

Freiheit der Wahl: jetzt gibt es die Alternative!

DM **798,-**
Versandkosten DM 6,-
DM **804,-**

Unverbindlich empfohlener Verkaufspreis

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Internationaler Vertrieb:
Compo Software GmbH
Tel.: 0 65 51 / 62 66

BESTELL-COUPON an Heim-Verlag Heidelberger Landstraße 194 6100 Darmstadt-Eberstadt

Ich bestelle ______ St. Publishing Partner MASTER à DM **804,-**
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei
zzgl. DM 6.– Versandkosten (Ausland DM 10.–), unabhängig von bestellter Stückzahl

Name, Vorname ______________________

Straße, Hausnr. ______________________

PLZ, Ort ______________________

Schweiz
Data Trade AG
Landstr. 1
CH-5415 Rieden-Baden

Österreich
Haider
Computer + Peripherie
Grazer Str. 63
A-2700 Wiener Neustadt

Tempus Word an Außergewöhnlichkeiten bietet (das wäre zuviel). Stattdessen werde ich die Möglichkeiten, die sich mit den Funktionen bieten, in den Vordergrund stellen.

Tempus Word verfolgt konsequent das Prinzip der grafischen Anzeige des Textes. Da mit verschiedenen Zeichensätzen gearbeitet wird, ist das auch notwendig. Von daher ähnelt es dem Prinzip von Signum!. Doch wie bereits angesprochen, ist beim Erfassen der Texte weniger Schönheit als Übersichtlichkeit gefragt. Bei Tempus Word gibt's noch einen ASCII-Modus, bei dem der Systemzeichensatz verwandt wird. Natürlich unproportional. Das Programm kann von der vollen grafischen Anzeige bis zur Kargheit des Tempus-Editors abgespeckt werden (siehe Bilder 2 und 3). Verwendet werden übrigens Fonts im eigenen Format. Signum!-Fonts können benutzt, allerdings nicht verändert werden.

Grundsätzliches

Wer - wie der Autor - bereits mit dem Tempus-Editor arbeitet, der wird zumindest mit dem ASCII-Modus keine Probleme haben. Sehr viele Tastenbefehle sind identisch. Die schnelle Eingabe der Texte stellt kein Problem dar. Ein satter Vorrat an Positionier- und Blockbefehlen, alle auch auf der Tastatur, läßt kaum einen Wunsch offen. Wer lieber im optisch reizvolleren 'Tempus Word-Modus' arbeitet, wird sich über den schlauen Cursor freuen. Der ist nämlich immer so breit wie der Buchstabe, auf dem er steht.

Wie steht's mit der Geschwindigkeit? Der Name Tempus steht ja für High Speed. Subjektiv würde ich sagen, daß im Grafikmodus Tempus Word ein wenig schneller ist als Signum!. Im ASCII-Modus muß man bedenken, daß der verwandte Zeichensatz immer als Berechnungsgrundlage für den Zeilenumbruch dient. Und formatiert wird bei Tempus Word ständig. Kommt dann noch die vollautomatische Trennung hinzu, wird erheblich gerechnet. Daher reicht die Geschwindigkeit nicht an die des Editors heran. Trotzdem - für eine Textverarbeitung ist es allemal sehr gut. Ein Nachlaufen findet nicht statt. Ich habe einige Zeiten gestoppt und zeige sie in der Tabelle. Das ist zwar nicht ganz fair (Script z.B. zeigt jeden Such/Ersetz-Vorgang am Schirm), mag aber als Anhaltspunkt dienen.

Das Konzept des Arbeitens im Grafik- oder Textmodus birgt aber Haken. Man möchte neben der einfachen Darstellungsweise auch eine ebenso einfache Druckweise nutzen können. Denn der klassische

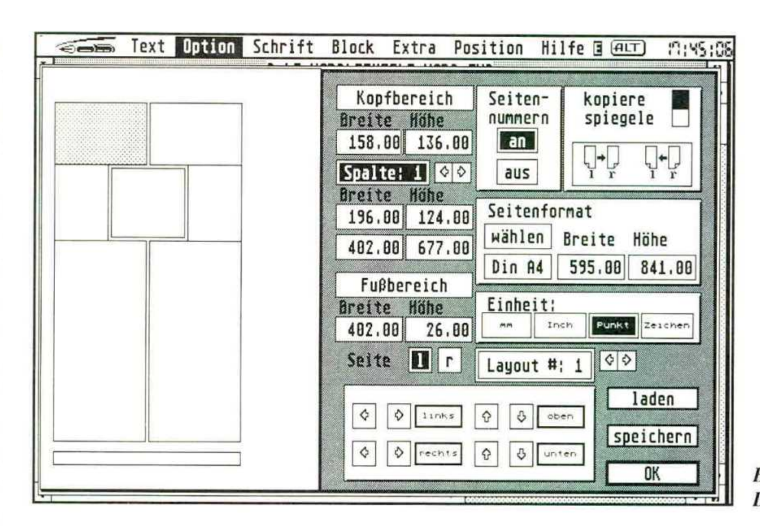


Bild 7: Luxuriös: Das Seiten-Layout

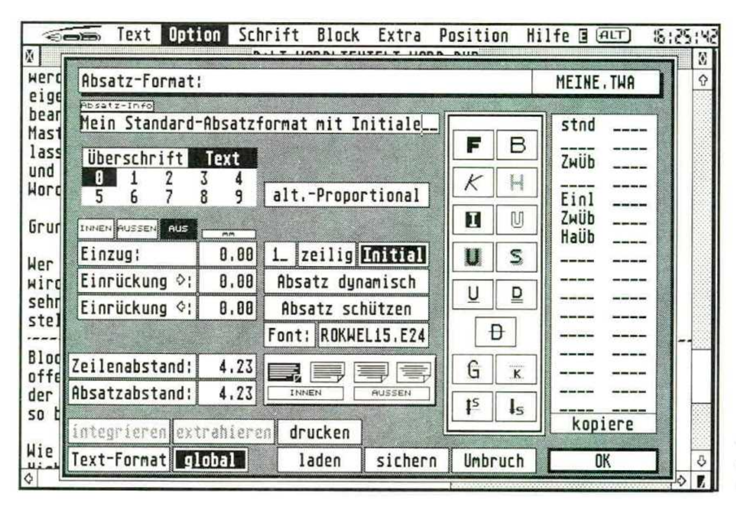


Bild 8: Absatz-Layouts für jeden Geschmack

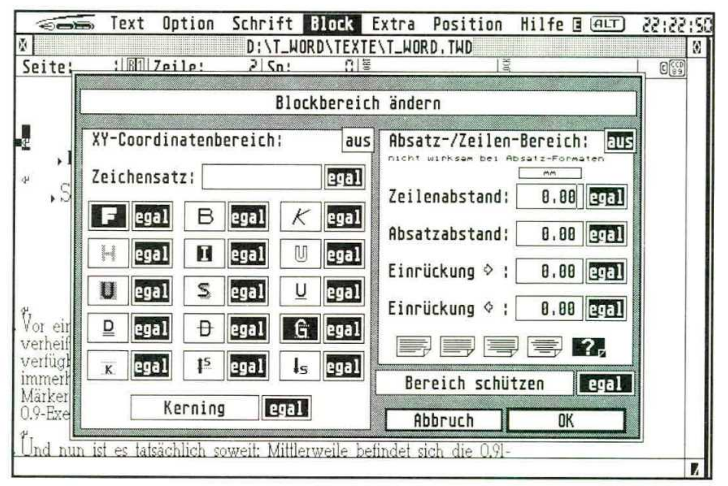


Bild 9: So bearbeitet man Blöcke.

ASCII-Druck mit den Fonts des Druckers ist um ein Vielfaches schneller als der grafische. Besonders bei Nadeldruckern. Nun haben aber die Zeichensätze des Druckers meist andere Zeichenbreiten als die Tempus Word-Fonts. Blocksatz in Proportionalschrift mit den Drucker-Fonts fällt also aus, oder?

Dem ist nicht so. Im Druckertreiber gibt man für alle Drucker-Fonts die Zeichenbreiten an. Lädt man den Treiber, ist Tempus Word in der Lage, den Text nach diesem Font umzubrechen, zeigt allerdings wahlweise den grafischen oder den System-Font an. Kompliziert, gell? Nach einigem Probieren blickt man aber durch. Der Vorteil dieser Zweiteilung liegt einfach darin, daß man sich Probedrucke in sehr guter Qualität erstellen kann, ohne ewig darauf warten zu müssen. Trotzdem stimmt das Druckbild - was die Positionierung des Textes betrifft - haargenau mit dem endgültigen Druck überein. Das verdient Lob.

Externa

Der Font-Editor ist ein ausgelagertes Programm. Sie sehen seine Oberfläche in Bild 4. Es liegt jedem Tempus Word-Font ein

Master-Font mit der doppelten Auflösung eines 24-Nadlers (720 DPI) zugrunde. Aus diesem errechnet der Editor in guter Qualität die Fonts für die verschiedenen Drucker und den Bildschirm. Dieses Konzept macht das Font-Format völlig inkompatibel zu allem üblichen, vor allem zu Signum!. Es sind außerdem in einem Tempus Word-Zeichensatz mehr Zeichen enthalten als in einem von Signum! Bleibt zu hoffen, daß es CCD bald schafft, mehr Fonts als die momentanen anzubieten. Denn mit deren Vielfalt steht und fällt die Gestaltbarkeit eines Textes.

In Bild 5 sehen Sie das Programm zur Erstellung der Druckertreiber. Es redet mit dem Anwender über ein simples Formular. Die Eingabe ist einfach, und es können einem kaum Fehler unterlaufen, zumal die lustigen Symbole geradezu motivierend wirken. Der Smiley 'compiliert' den Treiber, den man Tempus Word dann mundgerecht serviert.

Text-Styling

Ich werde nun einen neuen Text anlegen. Dann wird nicht nur ein Fenster geöffnet, und los geht's. Nein, zuerst müssen diverse Informationen über den Text und evtl. ein Paßwort eingegeben werden (Bild 6). Der Sinn dieser Informationen über Autor und Zweck nebst der Suchwörter zeigt sich dann, wenn man erst einmal die halbe Platte voller Texte hat. Dann ist jede Chance, ein bestimmtes Werk anhand seines Namens wiederzufinden, vertan. Auf dem Desktop von Tempus Word (Bild 1) ruft man eine Box auf, die von allen Texten eines Pfades jeweils nur diese Informationen zeigt. Dort kann dann auch nach Stichworten gesucht werden. Wie gesagt - für den Vielschreiber eine große Erleichterung.

Daran anschließend sollte man das Seiten-Layout festlegen (Bild 7). Auch hier präsentiert Tempus Word eine Dialogbox, die fast den Rahmen der Übersichtlichkeit sprengt. Doch wie bei so vielem: Man gewöhnt sich dran und sieht bald nur noch das Wesentliche. Leider trübt sich der Blick mit Wehmut, denn momentan sind nur einspaltige Layouts drin. Doch läßt die Box auf Mehrspaltensatz hoffen. Ein Hauch von DTP kommt auf. Apropos: Grafikeinbindung geht momentan auch nicht. Wer sich vor größerer Formatierarbeit bewahren will, der widmet sich jetzt schon den Absatzformaten (Bild 8). Diese Formate haben eine höhere Prioriät als das Seiten-Layout und auch als die globalen Einstellungen, die man mit Icons auf dem Desktop (warum eigentlich dort?) trifft. Die Sache mit den Absatzformaten hat gleich zwei gute Seiten:

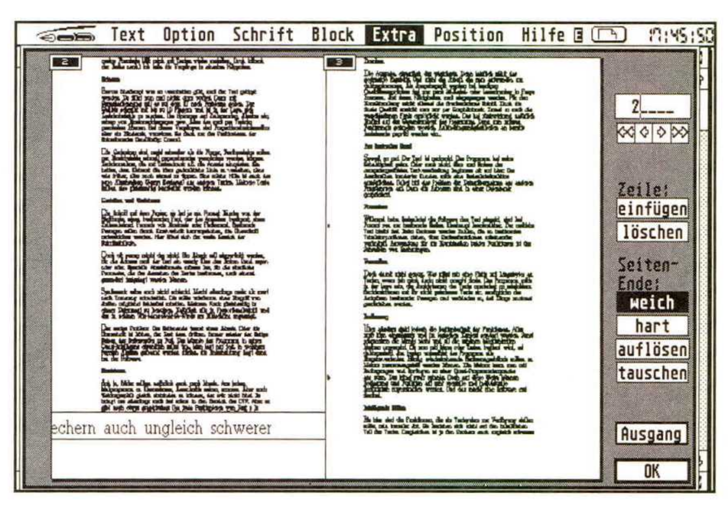


Bild 10: Ganzseitenansicht und Textlupe für den Umbruch

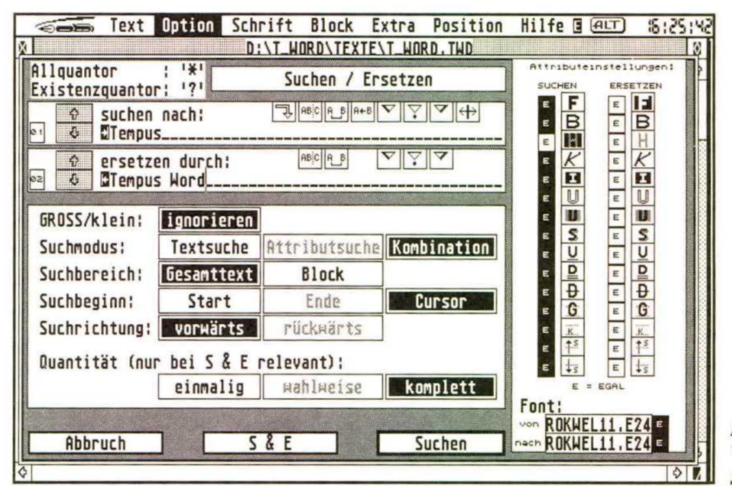


Bild 11: Läßt keine Wünsche offen: Suchen und Ersetzen

Erstens ist das einheitliche Formatieren eines Dokuments kein Thema mehr. Denn auf Tastendruck 'gießt' man den Absatz in eine festgelegte Form. Die Formate werden gespeichert und stehen so jedem Dokument (und jedem Autor!) zur Verfügung. Gleiches gilt natürlich auch für das Seiten-Layout. Arbeiten in einem Betrieb oder Institut mehrere Personen an einem Schriftstück, ist dessen gleichmäßige Form unproblematisch. Gefällt eine bestimmte Formatart nicht mehr, klickt man sich durch die Box, variiert sie ein wenig und formatiert den Text neu. Fertig.

Zweitens weiß das Programm auf diese Weise, ob ein Absatz eine Überschrift oder ein Text ist. Beides bekommt eine Gewichtung zwischen Null und Neun. Damit legt man fest, ob z.B. eine Überschrift nur auf einer ungeraden Seite oben stehen darf, oder auch mitten auf der Seite. Oder: Paßt beim Seitenumbruch die Überschrift noch auf die Seite, jedoch nur noch eine Zeile eines Absatzes von hoher Gewichtung, so wird alles auf die nächste Seite verfrachtet. Wer kennt sie nicht, die grausigen Überraschungen immer dann, wenn der Text mal mehr als 10 Seiten und man wieder etwas übersehen hat? Richtig dosiert kann Tempus Word helfen.

Oft genug reicht auch das Absatzformatieren nicht. Dann sollen einzelne Wörter hervorgehoben werden, z.B. durch doppelte Unterstreichungen. Tempus Word hat einiges Exotische auf Lager. Bei der mächtigen Blockformatierfunktion (Bild 9) bleibt kein Auge trocken: Attribute und Fonts werden gesetzt oder gelöscht oder unberührt gelassen - gerade so, wie man es braucht. So wie hier geht's einem oft mit Tempus Word: zuerst das undurchsichtige Chaos auf dem Schirm; hat man sich dran gewöhnt, dann eröffnen die so völlig parametrisierbaren Funktionen neue Dimensionen.

Weiter geht's

Nachdem mein Phantasietext, den ich gerade vor Ihren Augen bearbeite, nun die richtige Form hat, will ich ihn noch einmal komplett umbrechen. Dabei hilft mir die 'Textgestaltung'. Sie erledigt jetzt noch die Arbeit des Seitenumbruchs. Doch halt. Zwischen zwei ganz bestimmten Absätzen wollte ich eine freie Seite haben, denn dort soll später ein Bild hinein.

Also wähle ich 'Seitenumbruch edieren', und schon erscheint Bild 10. Die

Doppelseite ist verkleinert, und ich brauche nur zu klicken, wohin das harte Seitenende soll. Doch auf meinem 640x400-Monitor (Tempus Word läuft auch auf allen Großbildschirmen) seh' ich leider nicht viel. Das Autorenteam hilft. Der Text, über dem der Cursor gerade steht, wird per Lupe vergrößert. So ein Service und ein nützliches Detail.

Zur Verdeutlichung der Detailwut bei CCD ist im Bild 11 die Suchen/ Ersetzen-Dialogbox dargestellt. Es werden alle Passagen, an denen das Wort 'Tempus' auf einem linksbündigen Tabulator steht, ersetzt durch das Wort 'Tempus Word' auf einem rechtsbündigen Tabulator. Dabei werden nur die beachtet, bei denen mindestens das Atrribut 'Heller Text' gesetzt ist. Es wird in der Ersetzung 'Fett' gesetzt und alle anderen Attribute gelöscht. Noch Fragen?

In den Händen...

...halten möcht' ich ihn - den Text. Demnach steht jetzt das Drucken an. Über die verschiedenen Modi hab ich mich oben schon ausgelassen. Integriert ins Programm ist ein Spooler (Größe und Geschwindigkeit variabel, logisch!), in dem das Dokument erst einmal landet. Geduldet man sich mit dem Ergebnis, kann tatsächlich ganz gut nebenbei gearbeitet werden. Ach ja, gedruckt wird auf drei Ports: parallel, seriell und DMA. Und gleichzeitig. OK - ich hab' nur einen Drucker. Das Ergebnis ist beeindruckend, genauso gut wie Signum! halt. Aber mein P6plus hat wunderbare eingebaute Proportional-Fonts. Und mit denen kommt Tempus Word ja auch zurecht. Und es geht schneller. Und es sieht auch toll aus.

Apropos Signum!. Tempus Word - der Signum!-Killer? Die Frage stellen sich sicherlich viele. Zumal momentan der Preisunterschied nicht sehr groß ist. Die Features, die Tempus Word mal haben wird, lassen Signum! sicher verblassen. Bis auf das Definieren grafischer Blöcke und deren Bearbeitung gibt es keinen Unterschied. Die Mikroschritte sind drin, genauso wie die Zeichensätze. Darüber hinaus wird Tempus Word über eine ganze Reihe Mehr-Funktionen verfügen. Doch momentan hat Application Systems mit ihrem Produkt noch die Nase vorn. Denn sie bieten ein komplettes Programm. So elementare Dinge wie Fuß-/Endnoten, Grafikeinbindung aber auch Spaltensatz beherrscht Tempus Word noch nicht.

Zubehör

Neben all den reinen Hilfsmitteln beim Schreiben, Formatieren und Drucken reicht uns CCD noch Taschenrechner, Notizbuch, Lexikon und Wecker an die Hand. Fehlt nur die Lesestütze. Das Lexikon arbeitet wohl wie eine Dateiverwaltung, verwaltet aber auch Stichworte... In meiner Anleitung stand leider noch nichts darüber. Das Notizbuch ist ganz nützlich. Aber ich arbeite auch mal mit anderen Programmen, und da fehlt es dann. Dann muß ich doch Papier und Bleistift nehmen. Es sei denn, das Notizbuch wird eines Tages auch als Accessory zur Verfügung stehen...

Sehr nützlich sind viele der Details, auf die die CCD-Mannen so stolz sind: der variable Cursor, das Erstellen von Inhaltsverzeichnissen, die Ganzseitenanzeige usw. Eine On-Line-Hilfe spielt auf Tastendruck Hilfstexte zu den Funktionen ein. Acht Texte passen in den Speicher, ein einziger kann in vier Fenstern gleichzeitig dargestellt werden.

Vermißt habe ich die UNDO-Funktion, an die mich der Tempus-Editor doch sehr gewöhnt hat. Warum so häufig gebrauchte Menüpunkte wie 'Tempus-Word-Modus' oder 'Textgestaltung' keine Tastencodes erhielten - wer weiß. Ebenso unklar, warum sich das Programm einige Pfade nicht merkt. Die Trennfunktion mit Ausnahmelexikon funktioniert gut, die Trenntiefe ist leider nicht variabel. Mir ist es auch nicht gelungen, bereits gesetzte Trennungen aus einem Text zu entfernen. Kommen wir also zu ein paar Dingen, die der Version 0.91 fehlen.

Versprochen ist versprochen

Liebe Leute von CCD! Euer Programm ist schon ganz toll, doch bis zur Version 1.0, da wollen wir noch: Fuß-/Endnoten; Makroerstellung, denn einige Vorgänge erfordern dermaßen viele Tastendrücke, daß einem vor Dialog auf- und zuklappen schwindlig wird; Indexlistenerstellung; Seitenlayout, mehrspaltig und mit Kopf- und Fußzeilen; Grafikeinbindung; Serienbriefe; Formulare; Rechnen im Text. Die Gliederung ermöglicht das Einklappen von Absätzen ähnlich der Prozeduren bei GFA-BASIC 3.0.

Das sind keine Spinnereien von mir. Das verspricht uns CCD tatsächlich. Also wohlan und frischen Muts ans Werk!

Weiterführend ist ein Korrekturlexikon nötig. Vielleicht läßt sich auf dem ST auch eine On-Line-Korrektur realisieren? Und dann der Thesaurus. Davon haben alle etwas. Denn dann sind meine Elaborate nicht mehr gar so eintönig.

Ende gut, alles ...

Tja, nun bin ich wohl berufen, die Endwertung zu treffen. Also mal ganz im Ernst: Tempus Word ist ein begeisterndes Programm. Ich denke, nicht zu übertreiben, wenn ich die Lage auf dem Textverarbeitungssektor beim ST mit der Wasserversorgung in der Sahel-Zone vergleiche. Tempus Word ist dabei die Oase. Alle Funktionen, die momentan geboten werden, arbeiten sauber und schnell. Aber wie gesagt - da muß noch kräftig dran gearbeitet werden. Die Perspektive, die sich bietet, ist enorm: ein Programm, das über die reine Schreibmaschinenemulation mit Sonderausstattung hinausgeht. Vergleicht man die maximalen Möglichkeiten, die eingangs umrissen wurden, mit dem, was Tempus Word können wird: Hut ab. Ich kann damit Texte wirklich VERARBEITEN und nicht nur tippen. Der Vielschreiber wird Tempus Word nutzen; wer alle 14 Tage einen Brief schreibt, für den ist Tempus Word zuviel des Guten.

Bleibt letztlich nur die Frage, ob der Kunde als Beta-Tester der richtige Weg ist, Software an die Frau/den Mann zu bringen. Der Lohn für den Anwender beträgt 200 DM. Und früher mit Tempus Word arbeiten zu können.

IB

	Tempus Word 0.91	Tempus Editor 2.0	Signum! 2.0	1st_Wordplus 3.15	Script
Laden des Programms	64 s	17 s	29 s	22 s	33 s
Laden 35128 Bytes	9 s	5 s	7 s	18 s	14 s
Speichern 35128 Bytes	18 s	12 s	16 s	34 s	29 s
Scrollen 866 Zeilen	33 s / 24 s (*)	17 s	42 s	132 s	176 s
Ersetzen 3078 Zeichen	8 s	2 s	(**)	26 s	780 s

(*) Die zweite Zeit im ASCII-Modus (**) Signum! stürzte leider ab.

Kleiner Benchmarktest

Bezugsadresse:
CCD
Burgstr. 9
6228 Eltville
Tel. 06123/1638

SAMPLING

Sound Sampling Systeme in 8 Bit und 16 Bit

AS SOUND SAMPLER II Maxi +	AS SOUND SAMPLER III
8 Bit	16 Bit
HiFi Qualität	CD Qualität
Anti Alysing	Anti Alysing
11 – 36 kHz	20 – 45 kHz
max. 336 s*	max. 92 s*
Samplezeit netto	Samplezeit netto

Sample Software:**

Midifähig, Samplesequenzer, Transponieren, Loop, Echo, Fader, Auto Mixer mit Mix Editor, Reverse Play, Multisampling, Keyboard Editor, Sample Cutter mit null Position Finder, Soundbibliothekverwaltung, Sampleeinbindung in eigene Programme

Anschlußfertiges Gerät, Software und Handbuch
DM 298,—*** DM 598,—***

Zubehör: 10 Disketten, Soundbibliothek (Effekte)
DM 149,—*** DM 198,—***

* bei 4 MB Speicher
** Samplesoftware nur für Monochrommonitor
*** unverbindliche Preisempfehlung

G DATA

Siemensstr. 16
D–4630 Bochum 1
Telefonische Bestellungen:
0 23 23 / 38 98 58
Versand zzgl. DM 7,— Portopauschale per Nachnahme oder Vorausscheck

Schweiz:
DTZ Data Trade AG
Landstr. 1 • CH–5415 Rieden / Baden



Easybase

Datenbank-Leichtigkeit

Wenn die unbestrittene Nummer Eins in der Computeranwendung die Textverarbeitung ist, dann folgt auf dem zweiten Rang sicher das, was man unter dem Begriff "Datenbank" kennt. Gerade das Verwalten von großen Datenmengen, das schnelle Suchen und das ganz gezielte Sortieren und Ausgeben bestimmter Inhalte, das war doch schon immer die Wunschanwendung, mit der Programmierer allesamt die Gewalten unserer "Rechenknechte" ausnutzen wollten. So darf man durchaus mit Staunen vernehmen, daß das erfolgreichste, weil meistverkaufteste (und wahrscheinlich auch meist-raubkopierteste) Programm mit Familiennamen "dBASE" heißt. Selbst eingefleischte ATARI ST-Anwender kennen dieses Produkt und nicht nur alleine seinem Namen nach. Für den ATARI-Computer haben andere Programmnamen (die ich hier nicht wiederholen möchte) das Feld der Datenbank bereitet, auch einige "dBASE-Nachbauten" sind darunter.

Wenngleich die Auswahl an Datenbanken für den ATARI nicht gerade gering ist, werden dennoch (oder gerade deswegen?) immer wieder neue Datenverwaltungsprogramme vorgestellt. Und wenn es so aussieht, als hätten bekannte Namen das Feld (sprich die Marktanteile) längst abgesteckt, dann erscheint wieder eine Datenbank im Katalog der Neuankündigungen (wofür die Frühlings-CeBIT und die Spätsommer-ATARI-Messe hervorragend geeignet sind).

Mit in den Ring steigt ab sofort die Firma OMIKRON. Das Produkt der Software-Schmiede aus Birkenfeld bei Pforzheim heißt Easybase und darf wörtlich genommen werden! ("Nomen est omen" - wer kann Latein?)

Datenbank "light"

Uns lag zwar eine Vorabversion von Easybase vor, die war aber schon so "heiß" (Originalton), daß sicher keine Änderungen zur Verkaufsauflage mehr vorgenommen wurden ("irgendwann ist eben Schluß" - ebenfalls Originalton). Vom späteren Handbuch durfte ich einen Rohentwurf in Augenschein nehmen. Es wird auf jeden Fall einen durchschnittlichen "Bilderdurchsatz" von 1 Bild (also Hardcopy u.ä.) pro Seite haben (was als Faustregel für verständliche Handbücher gilt).

Nach dem obligatorischen Doppelklick erscheint eine ungewohnte Oberfläche: Die altbekannte Desktop-Leiste fehlt! Da Easybase wahlweise auch als Accessory installiert werden kann, mußte die obere Menüleiste wegen der Einheitlichkeit entfallen. Das bringt aber zwei Vorteile mit sich: 1. Easybase ist aus jedem anderen Programm aufrufbar (ok, Sie wissen sowieso, was ein Accessory ist), aber 2. sind die Desktop-Menüwörter (bei anderen Programmen) nicht immer sehr trefflich formuliert. Deswegen hat man sich bei Easybase für eine Icon-Leiste am unteren Bildrand entschieden, und Symbole sind nun einmal schneller zu interpretieren als Wörter.

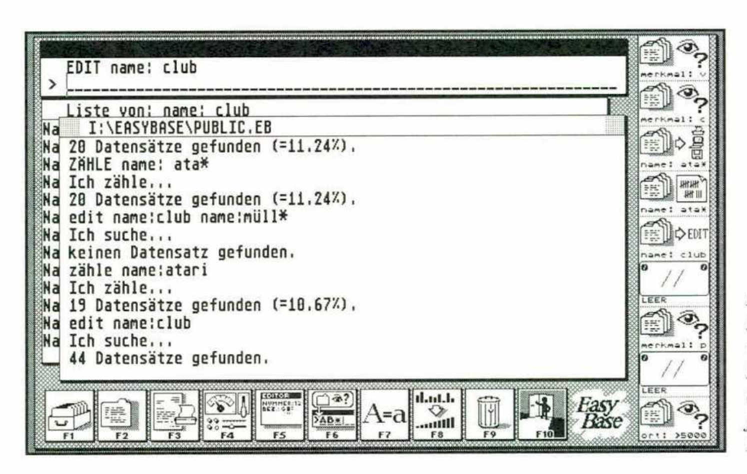


Bild 1: Das Hauptarbeitsfenster von Easybase zeigt die Kommandozeile (oben), Klemmbretter für häufig benutzte Befehlsfolgen

Am rechten Bildrand zieht sich eine Klemmbrettleiste nach unten, die Platz für 10 Befehlsfolgen hat. Dort werden also Such- oder Sortierdefinitionen, welche sehr häufig vorkommen, niedergelegt und brauchen fortan nicht mehr manuell eingegeben zu werden - Mausklick genügt (Wertung: sehr nützlich). Am oberen Bildrand hängt die Kommandozeile für die manuelle Befehlseingabe. Sie zeigt außerdem den zuletzt eingegebenen Befehl noch an. Mit den Tasten >Pfeil-hoch< bzw. >Pfeil-runter< rollt man nacheinander alle bisher eingegebenen (maximal

100) Befehlszeilen wieder in die Kommandozeile, um sie erneut ausführen zu lassen oder zu ändern. Anwender von dBASE kennen dies als "History-Buffer".

Den größten Raum des Bildes teilen sich ein Melde- und ein Listenfenster. Sie können abwechselnd in den Vordergrund "geklickt" werden, sind aber (leider) weder in der Größe, noch in der Position veränderbar. Auch hätte man sicher die Kommandozeile in das Meldefenster einbauen können, um somit eine bessere Bildaufteilung und etwas mehr Platz zu erhalten. Das Listenfenster besitzt zwar einen Rollbalken zum Umherwandern, allerdings auch nur in vertikaler Richtung. Somit sind von allen sichtbaren Datensätzen immer nur die Anfänge zu erkennen, der Rest hüllt sich in Schweigen. Um einen Datensatz in seiner kompletten Ausdehnung zu bewundern, genügt ein Doppelklick auf ihn. Im Meldefenster sucht man Rollbalken völlig vergeblich. Meldungen und Kommentare des Systems, die schon eine geraume Zeit zurückliegen, sind dadurch unerreichbar. Notlösung: Protokollausdruck auf den Drucker während der Datenbanknutzung. (Wertung: verbesserungsbedürftig).

Die Icons

Obwohl es eingangs plausibel erschien, auf die gute alte Desktop-Leiste zu verzichten, gibt es trotzdem Rollmenüs. Bei vier der Icon-Symbole rollen "Pop-Up-Menüs" (d.h. von unten kommend) auf. Bonbon am Rande: Das Menü ist sogar frei positionierbar und erscheint mit seiner linken oberen Ecke immer dort auf dem Bildschirm, wo sich gerade der Mauszeiger befindet. Zusätzlich können alle Icons und die Auswahlpunkte in den Menüs per Funktionstasten aktiviert werden. Andere vier Icon-Symbole lassen große Fenster erscheinen, wobei jene zum Editor- und zum Papierkorb-Menü mit viel Text überladen sind.

Ansonsten sind die Symbole sehr klug ausgedacht und einprägsam, so daß mit zunehmender Perfektion (des Anwenders) damit ein schnelles und gezieltes Arbeiten möglich wird.

Wie Easybase arbeitet

Wenn man die Datenbankprogramme von ihrem Arbeitsprinzip her sieht, gibt es eine ganze Menge unterschiedlicher Vorgehensweisen: hierarchische, relationale, assoziative, verkettete, vernetzte usw. "Wer braucht denn eigentlich relationale Datenbanken?" Diese Frage hört man öfters, besonders wenn gerne das Beispiel vom Karteikasten zur Verdeutlichung von Datenbankprogrammen beansprucht wird. Das Herstellen einer Relation mag für Anfänger schwierig zu durchschauen sein. Auch muß das Aussehen der Datenbank auf diese Relationalität aufgebaut sein, und Verhältnisse der Datenbanken untereinander müssen fest stehen.

Eines ist sicher: Easybase ist keine relationale Datenbank! Das Definieren einer Struktur mit Feldname, Feldlänge, Feldtyp, Schlüssel usw. ist nicht nötig! Es wird vielmehr das uralt bewährte Karteikartenprinzip umgesetzt. Man kann ruhig sagen, Easybase denkt wie ein Mensch, nämlich wortweise. Die meisten Datenbankprogramme sind streng zeichenorientiert und wollen z.B. bei einer Wortsuche die Folge von Buchstaben exakt vorgegeben haben: Der Herr 'Wilhelm Kaiser' ist ein anderer als 'Kaiser Wilhelm'. Easybase würde nun aber erkennen, daß von der Vorgabe "SUCHE Wilhelm Kaiser" alle Datensätze betroffen sind, die sowohl "Wilhelm" als auch "Kaiser" (z.B. Wilhelm P. Kaiser) enthalten. Selbst wenn Sie nicht mehr genau wissen, ob der Herr Kaiser mit Vornamen nur "Willi" hieß oder gar "Keyser" ganz anders geschrieben wird, Easybase kann solche Ähnlichkeiten herausfinden und in die engere Wahl bringen (Wertung: sehr nützlich).

Eine Datenbank definieren? - Easy

Wie wir schon festgestellt haben, wird in Easybase keine Datenbankstruktur vorgegeben. Das Programm kennt nur ein Kriterium: die Kategorie (vergleichbar mit Feldnamen). Eine Kategorie ist ein Gleichheitsmerkmal, also nichts anderes als ein Titel, der in jedem Datensatz dieselbe Stelle einnimmt. Dabei kann der Inhalt einer Kategorie beliebig lang sein ("beliebige Feld- und Satzlänge). Die jeweilig unterschiedlich langen Inhalte gleicher Kategorie werden dynamisch verwaltet, benutzen also immer nur den tatsächlich beanspruchten Speicherplatz. Maximal sind 1000 Kategorien möglich. Und von Datensatz zu Datensatz zeigt einfach ein Pointer auf dieselben Einträge, wodurch sich die Suchgeschwindigkeit wesentlich erhöht. Als Faustregel für die typische Suchgeschwindigkeit gilt: 1000 Adressen in 0,2 Sekunden. Ein wesentlicher Vorteil von Easybase ist zunächst das Präsenthalten der kompletten Datenbasis im RAM-Arbeitsspeicher. Dies bringt natürlich eine extrem hohe Verarbeitungsgeschwindigkeit mit sich, weil Lesezugriffe auf Diskette oder Festplatte im Programmablauf gänzlich unterbleiben. Im

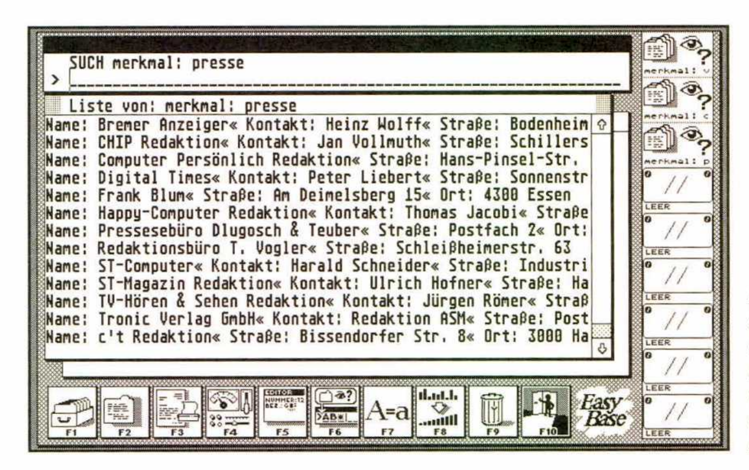


Bild 2: Nachdem in der Kommandozeile ein Suchbefehl eigegeben wurde, zeigt das Listenfenster (Bildmitte) das Ergebnis.

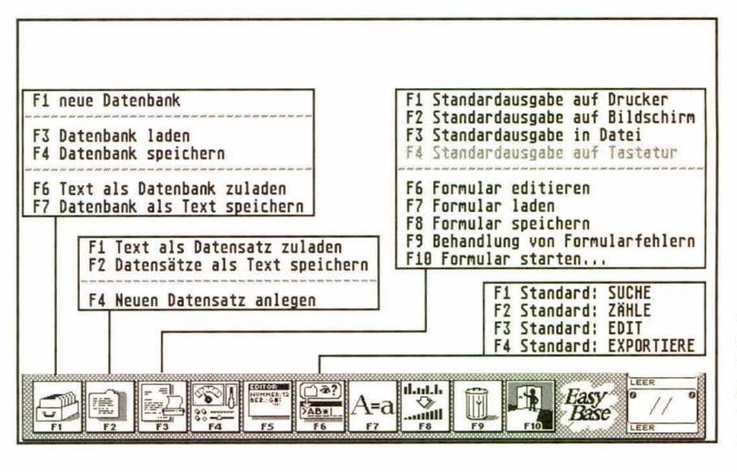


Bild 3: Einige Icons erzeugen sogenannte "Pop-Up-Menüs", wie sie der GEM-Profi von der Desktop-Leiste her kennt





NEU
Version 1.5

Einschreiber

That's Write - Junior

Textverarbeitung für Einsteiger

Neben dem Können der **Junior**-Version ist besonders der günstige Preis eine Entscheidung für alle, die **leicht** und **richtig** einsteigen wollen.

• **Flexibel,** d.h. einfacher Umgang mit einem komfortablen, **preisgünstigen** Textverarbeitungsprogramm. Jederzeit Aufstieg in die Profi-Version möglich

• **Graphikeinbindung**

• **Seitenlayout**

• **Absatzlayout**

• **Silbentrennung** nach deutschen Regeln, erweiterbar durch Eingabe eigener Wörter in unbegrenzter Höhe

• Bis zu 10 verschiedene **Fonts** (Schriftarten) in einem Text möglich

• Lieferung mit **10 Fonts (10 versch. Schriften gleich dabei)**

• **Ausdruck im Graphik- und Text-Modus** (auch gemischt innerhalb einer Zeile)

• **Blockfunktionen**

• **Suchen und ersetzen**

• **Tabulatoren**

• **ASCII einlesen und speichern**

• **Bilddarstellung entspricht dem Ausdruck** (Keine Eingabe von Steuerzeichen nötig)

• **Leicht verständliches deutsches Handbuch**

Preis: DM **148,-**

Aufstieg

Jederzeit **Aufstieg in die Profi-Version** möglich (Original-Diskette der Junior-Version an Heim-Verlag einsenden und Profi-Version bestellen)

Preis: DM **246,-**

Alle genannten Preise sind unverbindlich empfohlene Verkaufspreise.

Kundenbetreuung durch
Compo Software GmbH

Informationen
und
Hotline

Telefon: 0 65 51 - 62 66

Vielschreiber

That's Write

Die professionelle Textverarbeitung

Endlich! Ein Textverarbeitungsprogramm, das neben seinem Können auch anwenderfreundlich ist.

That's Write ist für alle gängigen Anwendungen einer Textverarbeitung geeignet. Dank der einfachen Bedienung per Maus ist es für den Einsteiger leicht und schnell zu erlernen. Dem viel schreibenden Anwender bietet es die Bedienung über die Tastatur oder die Definition eigener Funktionen über Makros.

Komplettpaket mit großem Funktionsumfang:
• eigener Fonteditor • internationale mehrfach frei belegbare Tastatur • alle Tasten mit Floskeln und Makros belegbar • zuverlässige eingebaute Silbentrennung • umschaltbare Trennregeln für verschiedene Sprachen • eingebaute Rechtschreibkorrektur • das Hauptwörterbuch kann gewechselt werden (Option Englisch, Niederländisch, weitere in Vorbereitung) • eingebaute Serienbrieffunktion mit Schnittstelle zu Datenbanken und Adressverwaltungen • integrierte Schnittstelle zu Accessories, Datenübernahme aus Adressverwaltung • autom. Erstellung eines Stichwortverzeichnises • autom. Erstellen eines Inhaltsverzeichnises • autom. Fußnotenverwaltung • autom. Endnotenverwaltung • Umbenennen, kopieren, löschen von Dateien vom Programm aus • Snapshot-Accessory erlaubt aus anderen Programmen Bilder vom Monitor abzugreifen und diese in That's Write zu verwenden • einfache Installation • Gliederungsfunktion

Optimale Druckerausnutzung:
• 9-Nadel-Drucker (alle Auflösungen) • 24-Nadel-Drucker (alle Auflösungen, auch 360x360 dpi) • Laserdrucker (alle Auflösungen) • Atari-Laserdrucker SLM804 über DMA
• Nur Textdruck bei Typenraddruckern (auch Proportionalschrift in Blocksatz!), verschiedene Typenräder anpaßbar

Ausdruck von Text- und Graphiktextmodus in einer Zeile:
• optimaler Randausgleich • unterstützt interne Schriften des Druckers, wie Proportional, Pica, Elite, schmal, breit, doppelt hoch • unterstützt Grafikfonts freier Größe • unterstützt beliebig viele Download-Fonts in einem Text • unterstützt druckerinterne Vektorfonts (Kyocera u.a.)

Durchdachte Bedienung:
Erstmalig ein übersichtliches und schnelles Bearbeiten von selbst langen Texten. z.B.: „Text zuklappen" versteckt in Sekundenbruchteilen den normalen Text – und zeigt nur noch die Kapitelüberschriften. Dort geht man schnell in das gewünschte Kapitel und „klappt" wieder auf. Noch schneller geht es wohl wirklich nicht mehr.

Einzigartiges Druckkonzept:
That's Write druckt Grafiktextmodus und Textmodus gleichzeitig! Wer bietet mehr?

Optimierte Seiteneinteilung beseitigt drei klassische Probleme automatisch:
• Umbruch zwischen Überschrift und Text wird automatisch verhindert
• Leerraum nach Seitenumbruch wird nicht ausgedruckt, d.h. gleichmäßiger oberer Rand
• Einzelne Zeilen eines Absatzes alleine auf einer Seite werden automatisch verhindert.

Bildschirmanzeige entspricht dem Ausdruck (WYSIWYG)

Einfachste Bedienung per Maus, Tastatur oder Makros

Flexible Textgestaltung durch Absatz- und Seitenlayouts

Weitere Vorzüge:
• Unterstützt selbständig den Blitter • läuft unter allen bekannten TOS-Versionen und unter GEM 2.2 • unterstützt Farb- und Monochrombildschirm sowie Großbildschirme • korrekte Behandlung von getrennten Wörtern bei Suchen, Ersetzen und Rechtschreibekorrektur • zwei Texte gleichzeitig bearbeitbar, schnelles Wechseln zwischen den Texten • autom. Sichern während des Schreibens in einem einstellbaren Zeitintervall • Einfügen von Kurz-Datum, Lang-Datum, Seitennummer, Folgeseite per Tastendruck • Textstatistik – Anzahl der Wörter, Zeilen, Seiten, Bilder • Zählfunktion für Wörter (und/oder Wort-Teile) • Ausschneiden/Einfügen von Blöcken mit 4 unabhängigen Puffern • Einfüge- und Überschreibmodus auch bei Proportionalschrift • flexibles ASCII-Laden/Sichern zeilen/absatzweise • Textübernahme von 1st Word/Wordplus. Vorhandene Texte werden mit Fußnoten und allen Attributen wie fett, kursiv, hoch/tief, unterstrichen eingelesen • 10 Marken zum schnellen Anspringen von Textpassagen, die mit dem Text abgespeichert werden • Hilfsfunktionen wie „Wort klein schreiben", „Wort groß schreiben", „1. Buchstabe groß/Rest klein" zur schnellen Korrektur von Tippfehlern. • komfortables Bewegen im Text: zeichenweise, wortweise, zeilenweise, absatzweise, Bildschirm-seitenweise Textseitenweise (zur Kontrolle des Seitenumbruchs), Text Anfang/Ende, Block Anfang/Ende, Seitenanfang
• Druckausgabe auf Druckerport (schnelle Direktansteuerung), Druckerport über Bios, RS232 oder Datei – Bilder können beim Drucken auch weggelassen werden (Probeausdrucke, usw.) • So lassen sich zuerst „gerade" und später „ungerade" Seiten ausdrucken, um Papier komfortabel beidseitig zu bedrukken, ohne jedes Blatt einzeln wenden zu müssen
• Hotline für registrierte Anwender

Preis: DM **348,-**

Änderungen von That's Write 1.3 auf Version 1.5:

1. Rechtschreibkorrektur nach Langenscheidt:
   a) Das deutsche Wörterbuch kennt ca. 3 Millionen Wörter
   b) Trennung nach Wörterbuch möglich, daher erstmalig auch korrekte Trennung von zusammengesetzten Wörtern: „Gas-tank", „Hals-tuch"
   c) Fehler bei Groß-/Kleinschreibung werden, wenn eindeutig, automatisch erkannt und behoben!
   d) Sinnvolle Korrekturvorschläge, so wird z.B. bei „Füsiker" nur das Wort „Physiker" vorgeschlagen.
   e) Einfaches Erweitern und Bearbeiten von Zusatzwörterbüchern und Trennungen.
   f) Zusätzliche Wörterbücher erhältlich: Französisch, Englisch, Amerikanisch, Spanisch, Schwedisch, Schweizer-Deutsch
2. Textverkettung – jetzt kann per Knopfdruck in den folgenden oder den vorigen Text gewechselt werden. Seitennummern und Fußnotennummern werden automatisch übernommen
3. Fonts löschen, Anschlagzählung
4. Paßwortschutz verhindert unberechtigtes Lesen von neugierigen, naseweisen Wesen
5. Textinfo speichert zum Text: Autor, Stichworte, Erstellungsdatum
6. Doppelseitiges Drucken, da linke und rechte Kopftexte (Heftrand kann berücksichtigt werden)

• That's Write Junior
• That's Write Profi
jeweils einschließlich
Signum* Font-Konverter

Weitere
That's Write Font-Disketten
lieferbar

Die Textverarbeitung



Preisgünstig ☑
Bedienungsfreundlich ☑
Professionell ☑

Massenschreiber

That's Write - PS *

Erste PostScript*-fähige Textverarbeitung

• **Datenübergabe** an Fotosatz und Belichtungsstudios möglich. Reproqualität: 2000 dpi und mehr

• **Mitgelieferte Fonts:**
UTimes (Größen von 8-36 Punkt)
UHelvetica (Größen von 8-36 Punkt)
UDingbats

• **Ausgereifte Schriften**

• **Optimaler Qualitätsdruck** von Text und Bildern

• Drucken über UltraScript* oder ähnliche Programme

• Alle Funktionen wie in That's Write Profi

Preis: DM **598,-**

Aufstieg

Für registrierte Anwender:

Jederzeit **Aufstieg** von der Profi-Version That's Write **in die That's Write PostScript*-Version** möglich. Einfach That's Write PostScript*-Version als Aufstieg bestellen.

Preis: DM **298,-**

That's Write - PS Plus

Wird **zusätzlich** zur That's Write PS*-Version **mit UltraScript*** geliefert. Damit kann man beliebige PostScript-Dateien auf einem SLM 804, Laserjet, Deskjet (schneller Ausdruck durch Datenkompression) oder auch auf Matrixdruckern ausgeben. Zum Lieferumfang gehören u.a. die Schriften Times*, Helvetica*, Courier* und Symbol*.

Preis: DM **898,-**

UltraScript *

PostScript * auf Ihrem Drucker

Für registrierte Anwender von **That's Write PostScript*** besteht die Möglichkeit, das Programm **mit** einer speziellen Version von **UltraScript*** zu **erweitern**. Diese Version beinhaltet **Treiber für alle gängigen Drucker** (9 u. 24 Nadeln, sowie ATARI Laserdrucker) und die Fonts Times*, Helvetica*, Courier*, Lucida Font Family*

Preis: DM **398,-**

Adresschreiber

That's Address

Die Adressverwaltung zu That's Write

Direkte Übernahme einer Adresse von That's Address in einen Brief, ohne die Daten noch einmal schreiben zu müssen oder das Programm zu verlassen. Aus diesen Adressen nach einigen Kriterien **bestimmte Einträge für einen Serienbrief selektieren**. Unter Berücksichtigung dieser Vorgaben für That's Write haben wir That's Address entwickelt:
Ständig verfügbar: That's Address ist ein Accessory, d.h. aus einem laufenden Programm kann auf die Daten zugegriffen werden. Das Programm wird beim Einschalten des Rechners einmal automatisch gestartet und bleibt bis zum Ausschalten verfügbar. Oder als Programm: That's Address läuft auch als Programm – statt dem Accessory kann es (bei zu geringem Speicherplatz) als Programm gestartet werden.
Hohe Datensicherheit: Jede Änderung oder Neuaufnahme einer Adresse wird sofort abgespeichert. Einfache Bedienung, wahlweise per Maus oder Tastatur lassen sich alle Funktionen betätigen.
Adressübergabe an That's Write: Wählen Sie den Knopf „Übergabe" – Sie wechseln automatisch zurück in die Textverarbeitung, und die angezeigte Adresse erscheint an der gewünschten Stelle im Text.
Serienbrief/Datenexport: Ausgewählte Adressen können an That's Write übergeben werden. In Verbindung mit einem von Ihnen gestalteten Text macht That's Write daraus ein Rundschreiben.
Selektion: Adressen können nach freien Kriterien ausgewählt werden. Ausgewählte Listen können sofort bearbeitet werden oder zur späteren oder mehrfachen Verwendung (z.B. erst bearbeiten, dann Etikett, Serienbrief) abgespeichert werden. Manuelle Selektion: Einzelne Adressen können von Hand (auch mehrfach) z.B. für Etikettendruck abgespeichert werden.
Formulardruck: That's Address bedruckt Briefumschläge, Karteikarten, Überweisungen mit beliebigem Absender/Empfänger. Listen-/Etikettendruck: Ausgewählte Adressen können auf dem Drucker als Liste/Endlos-Etiketten ausgegeben werden. Formularanpassung: Die Formulare wie Überweisung, etc. können über That's Write von jedem selbst geändert werden. Übersichtlicher Bildschirmaufbau: Alle Feldbeschriftungen wie Straße, PLZ, Ort sind in kleiner Schrift gehalten (man kennt die Felder nach mehrmaliger Benutzung sowieso auswendig) und können vom Benutzer einfach geändert werden. Alle Daten wie „Peter Müller", „Dorfstr. 34" sind in normaler Schrift gehalten. Dadurch wird der Bildschirmaufbau nicht überladen und wichtige Daten sind auf einen Blick zu erkennen.
Für den Benutzer von That's Address empfehlen wir einen ATARI ST mit 1 MB RAM (läuft ohne That's Write notfalls auch mit 512 kB).

Preis: DM **189,-**

Freizeit – Funzeit

FunFace

Das Gesichtsmenü
5.576.787.923.200.000 Gesichter

Quick Brown Fox ® 1989

If he knew of this before,
the world surely would look a lot more fun











Läuft auf allen ATARI ST/Mega in hoher und mittlerer Auflösung (auch auf Stacy) • Schieben Teile links, rechts, auf, ab, nach innen und außen • Auswahl per Ziffer oder Zufallsgenerator • Zufallsanimation • Sichern und Laden als FAC • Grafikmenü: Freihand, Sprühdose & Füller, mit Muster • Primitives • Text • Vergrößern • Ausschneiden, Kopieren & Einfügen von/nach Clipboard • Outline • Bas-Relief • Spiegeln & Flip-Flop • Abdunkeln & Aufhellen • Pixelate & Raster • Stauchen & Strecken • Schatten & Multischatten • 5 Bildschirme • Sichern/Laden von IMG, PIx, RGH, CLP & ICN

Preis: DM **128,-**

* Erwähnte Warenzeichen oder Handelsmarken: PostScript (Adobe), Signum (Application Systems), UltraScript (QMS inc.), Lucida (Bigelow & Holmes), Times, Courier, Helvetica (Linotype AG)

Presented by Compo Software

☐ Info-Broschüre zu That's Write – kostenfrei

BESTELL-COUPON

Bitte senden Sie mir:

zzgl. Versandkosten DM 6,- (Ausland DM 10,-) unabhängig von der bestellten Stückzahl

	Produkt	Preis
______	That's Write Junior	à 148,- DM
______	That's Write Profi	à 348,- DM
______	That's Write PostScript*	à 598,- DM
______	That's Write PostScript* Plus	à 898,- DM
______	Aufstieg Junior V. auf Profi V.	à 246,- DM
______	Aufstieg Profi V. auf PostScript* V.	à 298,- DM
______	That's FunFace	à 128,- DM
______	UltraScript*	à 398,- DM
______	That's Address	à 189,- DM

Name, Vorname ______________________

Straße, Hausnr. ______________________

PLZ, Ort ______________________

Oder benutzen Sie die in ST-Computer eingeheftete Bestellkarte

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

In der Schweiz:
an Data Trade AG
Landstr. 1
CH - 5415 Rieden-Baden

In Österreich:
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

RAM eines MEGA-ST4 hätten etwa 40.000 Standardanschriften (Name, Vorname, Straße, PLZ, Ort, Telefonnummer) Platz. Indexlisten müssen nicht verwaltet werden, weil die Kategorien mittels Zeiger (Pointer) verkettet sind. Gerade die Schlüssellisten aber sind es, welche den konventionellen Datenbanken die Masse an Speicher rauben. Ja, und wie wird nun eine Datenbank definiert?

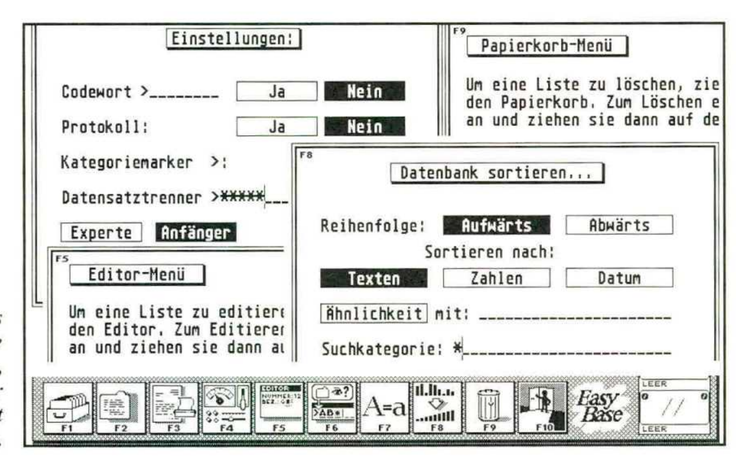


Bild 4: Einige Icons lösen umfangreiche Auswahlfenster aus, die manchmal sehr ausführlichen Text enthalten.

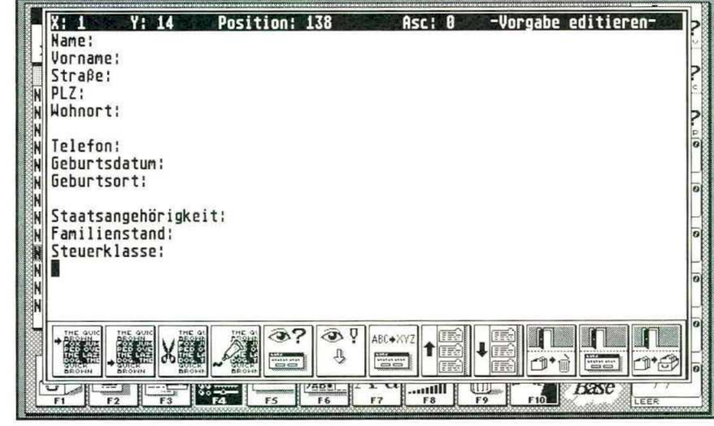


Bild 5: Da es bei Easybase keine Masken gibt, werden die (Pseudo)-Feldnamen im Editor einfach als Text geschrieben.

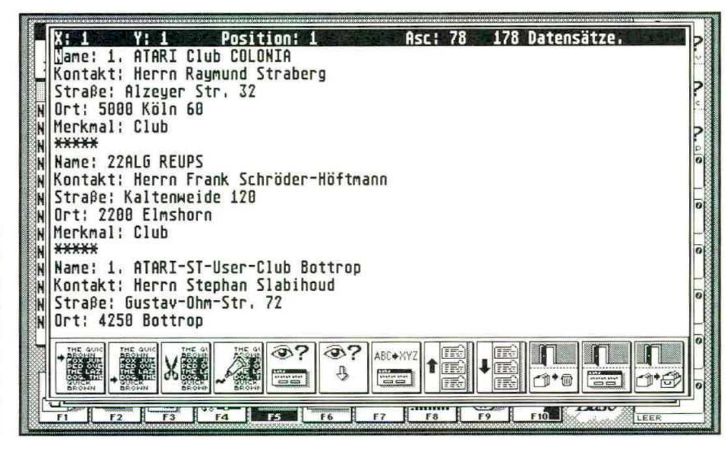


Bild 6: Das Ergebnis einer Suche oder Auswahl kann ganz oder teilweise in den Editor projiziert werden, wodurch nun komplette Datensatzinhalte sichtbar sind.

Der Editor

Bei allen Arbeiten innerhalb der Datenbank, besonders jenen, die verändernd auf die Inhalte einwirken (also nicht suchen, zählen usw.), wird ein Editor zugeschaltet. Er dient anfangs zur Definition (hauptsächlich Position) der Kategoriebezeichnungen. Diese schreibt man einfach von oben nach unten, ohne besondere Vorgehensweisen beachten zu müssen (eben wie ganz normalen Text). Selbst wenn die Anordnung der Kategorien schon feststeht, und sogar wenn schon Daten eingetragen sind, ein Hinzufügen und Löschen von Kategorien ist jederzeit im Editor möglich.

Beispiel: In einer Kundendatei soll neben Name, Vorname, Straße, Ort für einige Kunden noch die Telefonnummer eingetragen sein. Bei herkömmlichen Datenbanken müßte jetzt eine erweiterte Maske angelegt werden, die diesen Eintrag für alle Datensätze vorsieht. Damit wäre die neue Datenbankstruktur noch nicht verwirklicht, also ein Reorganisationslauf ist nötig. Dabei werden alle Datensätze an jenen Stellen aufgetrennt, wo nunmehr die Telefonnummer hinzukommen soll. Ein sehr aufwendiges Verfahren. Easybase verhält sich bei den besagten zusätzlichen Telefonnummern so, als würde man auf Karteikarten (und nur bei denen, wo es nötig ist) zusätzlich den Vermerk der Telefonnummer anbringen.

Der Editor dient schließlich, und zwar hauptsächlich zur Aufnahme der Ergebnisse irgendwelcher Suchprozesse. Als Feinheit in der Suche wäre hervorzuheben, daß Easybase auch unter Mehr-Wort-Inhalten gezielt z.B. nur das 2. Wort herausziehen kann.

Rechendefinitionen sind zunächst nur beim Export von Daten möglich. Grundgedanke war es, daß irgendwelche Rechenoperationen oftmals nur bei der Ausgabe (z.B. Listendruck) nötig sind. Beispiel: "Brutto:{[Netto]*1,14}" Diese Operation bringt in die Kategorie "Brutto" den Inhalt der Kategorie "Netto" mal 1,14, also plus 14%. Vorschlag von mir: Wie wäre es mit Rechnen direkt in der Datenbank? Trotzdem, die Rechenoperationen selbst können sich sehen lassen: Grundrechenarten, Potenz, Exponential, Logarithmus, trigonometrische und hyperbolische Funktionen, und das noch mit beliebig tiefer Klammerschachtelung.

Ein umfangreicher Exporteur

Was nützt es, die vielfältigen Funktionen auf dem Bildschirm ablaufen zu sehen? Gerade im Datenaustausch mit anderen Programmen sowie im Ausdruck von Listen, Etiketten oder Serienbriefen erweist es sich, wie komfortabel eine Datenbank wirklich ist.

Easybase bietet die Möglichkeit, mit anderen Programmen, die ACSII-Daten importieren können, zu korrespondieren (z.B. Textverarbeitungen). Außerdem reicht der Easybase-Editor zum Erstellen von einfachen Exportformularen (z.B. für Serienbriefe) völlig aus. Die Zeile im Editor: "Sehr geehrte [Anrede] [Name]" macht während des Exports auf den Drukker daraus z.B.: "Sehr geehrte Frau Müller-Thurgau".

Sehr interessant wird es, wenn man Easybase als Accessory installiert und es aus einem anderen Anwendungsprogramm (z.B. Wordplus) aufruft. Dann können Daten auch direkt in den Tastaturpuffer exportiert werden, was zur Folge hat, daß sie wie von Geisterhand direkt in die laufende Anwendung geschrieben

werden. So kann man z.B. Adressen speichern und dann als Briefkopf in anderen Programmen verwenden.

Der Versuch eines Überblicks

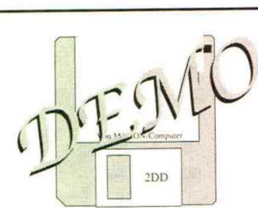
Easybase ist ein Datenbankprogramm, das neue Wege geht. Sehr schön ist das Losgelöstsein von festen Strukturen und Grenzen. Das Erstellen einer Datenbasis geht extrem schnell und einfach. An die Arbeitsweise mit den Icons gewöhnt man sich bald, auch die Klemmleiste rechts ist für häufig benutzte Befehlsfolgen sehr nützlich. In punkto Geschwindigkeit erscheint mir Easybase sehr flott (dies ist aber mangels Vergleich eine gefühlsmäßige Einschätzung). Ob die Tatsache, daß Easybase nur im RAM arbeitet, an schnelle Aufnahmegrenzen stößt, konnte ich nicht feststellen (mangels ausreichend großer Datenmenge). Vielleicht hilft bei dieser Überlegung jene Tatsache, daß das Neue Testament komplett Platz in Easybase gefunden hat. Viele Kleinigkeiten und Feinheiten, z.B. bei der Definition von Suchbedingungen, zeigen die Stärken von Easybase auf, z.B. 'Such nach ähnlichen Wörtern'. Mir hat die Arbeit mit Easybase gefallen. Für DM 248,- bewegt sich das Programm in vertretbarem Preisrahmen und dürfte in der Konkurrenz nicht nur deswegen ein Wörtchen mitzusprechen haben.

DK

Bezugsquelle:

OMIKRON Soft + Hardware GmbH
Erlachstraße 15
7534 Birkenfeld
Telefon: 07082/50048







Demo-Disketten

Damit Sie nicht immer die Katze im Sack kaufen müssen, haben wir ab sofort eine neue Rubrik für Sie eingeführt; es sind die Demo-Disketten von kommerzieller Software. Sie kosten lediglich DM 10,- pro Diskette und können über die Redaktion bezogen werden. So müssen Sie zum Vergleich verschiedener Programme nicht an verschiedene Hersteller schreiben, sondern können sich in aller Ruhe das Demonstrationsprogramm ansehen, bevor Sie das Original kaufen.

Bitte beachten Sie, daß die angebotenen Disketten nur Demonstrationsdisketten der Originalversionen sind und somit im Gegensatz zu den Originalen im Funktion eingeschränkt sind!

Folgende 25 Demo-Disketten sind z.Zt. erhältlich:

D1: S.&P.-Charts
Chart-Analyseprogramm
(*S.P.S. Software*)

D2: SPC-Modula-2
Modula-2-Entwicklungssystem
(*Advanced Applications Viczena*)

D3: ST-Fibu
Finanzbuchhaltungsprogramm
(*GMa-Soft*)

D4: ST-Fibu-Fakt
Fakturierungsprogramm für ST-Fibu
(*GMa-Soft*)

D5: ST-Fibu-Text
Textverarbeitungsprogramm für ST-Fibu mit Serienbrieffunktion
(*GMa-Soft*)

D6: SciGraph
Programm zur Erstellung von Präsentationsgrafiken
(*SciLab GmbH*)

D7: ST-Statistik
Uni- und multivariates Statistikprogramm, Grafikeinbindung (*SciLab GmbH*)

D8: fibuSTAT
Finanzbuchhaltungs-/Statistikprogramm
(*novoPLAN Software GmbH*)

D9: Btx/Vtx-Manager
Programm zum Anschluß an Bildschirmtext
(*Drews Btx + EDV GmbH*)

D10: Edison
Editor für fast alle Gelegenheiten
(*Kniss Soft*)

D11 & D12: CADjA
CAD-Programm für hohe Ansprüche
(*Computer Technik Kieckbusch*).
Demo besteht aus zwei Disketten zu je DM 10,-!

D13: JAMES 2.0
Programm für Börsenspekulanten
(*IFA-Köln*)

D14: Soundmerlin
Sample-Editor-Programm mit vielen Modulen
(*TommySoftware*)

D15: Soundmachine II
Programm zur Erstellung und Wiedergabe von Sounds
(*TommySoftware*)

D16: ReProK
Büroorganisationsprogramm
(*Stage Microsystems*)

D17: Sherlook
Schrifterkennungs- und -verarbeitungsprogramm
(*H.Richter*)

D18: ST Matlab
Programmiersystem mit Schnittstelle zu Modula-2
(*Advanced Aplications Viczena*)

D19: Calamus
Desktop-Publishing-Programm
(*DMC*)

D20: GD-Fibu
Finanzbuchhaltungsprogramm
(*GDAT*)

D21: Omikron.Draw!
Zeichen- und Malprogramm
(*Omikron.Software*)

D22: Omikron.Libraries
Verschiedene Libraries für Omikron.BASIC
(*Omikron.Software*)

D23: Omikron.Compiler
Demo-Version des Omikron.BASIC-Compilers
(*Omikron.Software*)

D24: Mortimer
Multi-Programm für alle Gelegenheiten
(*Omikron.Software*)

D25: Script
Textverarbeitungsprogramm
(*Application Systems /// Heidelberg*)

D26: SuperScore
Sequencer- und Notendruckprogramm
(*BELA Computer GmbH*)

D27: SPS-Emulator
Programm zur Simulation von SPS-Steuerungen
(*Karstein Datentechnik*)

D28: STAD 1.3+
Zeichenprogramm mit 3D-Teil
(*Application Systems /// Heidelberg*)

D29: MegaFakt
Fakturierungsprogramm
(*MegaTeam*)

D30 & D31: MegaPaint II
Zeichenprogramm mit Vektorteil
(*TommySoftware*)

D32: Tempus Word
Textverarbeitung
(*CCD*)

D33: Creator
Zeichenprogramm mit Animationsteil
(*Application Systems /// Heidelberg*)

D34: Outline Art
Utility für Calamus
(*DMC*)

D35: compugraphic Schriften
für Calamus
(*DMC*)

D36: BTX-Börsen-Manager
Börsenprogramm
(*Thomas Bopp Softwarevertrieb*)

Es gelten die gleichen Vertriebsbedingungen wie für PD-Disketten (s. PD-Seiten am Ende dieser Ausgabe). Demo-Disketten können auch zusammen mit PD- und Sonder-Disketten bestellt werden.

Bitte vergessen Sie nicht die betreffende Bestellnummer (z.B. D1) anzugeben.

Quick Dialog

Dialoge leichtgemacht

Dialoge sind das wichtigste Ein- und Ausgabemedium des ATARI ST, denn ohne sie ließen sich viele Programme nicht so komfortabel bedienen. Zudem erleichtern RSC-Construction Sets die Erstellung in hohem Maße. Die so erzeugten Dialoge lassen sich leicht einbinden, haben aber manchmal leider auch Nachteile. So kann ein RSC-File auch von "bösen Buben" verändert werden, um zum Beispiel die Copyright-Meldung zu löschen. Oder man vergißt beim Kopieren dieses File.

Deshalb gibt es für die Sprache C die Möglichkeit, RSC-Files einzubinden, in Pascal war dies bisher leider nicht möglich.

Dafür ermöglicht ST-Pascal+ die einfache Erstellung von Dialogen durch eigene Befehle. Nachteilig erweist sich dabei nur, daß zum Anschauen eines solchen Dialoges erst compiliert werden muß. Bei meheren Änderungen an einem solchen Dialog kann dies schon sehr lästig werden.

Um die oben angeführten Mankos zu beheben, gibt es nun von CCD das Programm Quick Dialog.

Ausgeliefert wird es in einer festen Box, in der sich neben der Programmdiskette noch ein "Handbuch" befindet.

Dieses ist mit einem Umfang von 28 Seiten auf den ersten Blick sehr kurz geraten, ist aber ausführlich genug, wenn man schon Erfahrung in der Erstellung in Pascal hat. Bei weitergehenden Informationen wird auf das ST-Pascal Handbuch verwiesen. Die Programmfunktionen werden aber genau erklärt.

Nun aber zum Programm selbst. Auf der Diskette liegt Quick Dialog sowohl als Accessory wie auch als Programm vor. Somit ist es möglich, auch während des Programmierens jederzeit Quick Dialog einsetzen zu können, ohne ST-Pascal verlassen zu müssen. Nach dem Programmstart darf man sich entscheiden, ob man Dialoge oder Alertboxen konstruieren möchte.

Fangen wir mit letzterem an. Quick Dialog bietet hier einen Dialog an, der Alertboxen zum Kinderspiel macht (siehe Bild 1).



Bild 1: Einfache Konstruktion einer Alertbox

Die verschiedenen Icons können angewählt werden und auch die Textlänge für den Boxtext und die Buttons ist vorgegeben, so daß es keine Probleme mehr gibt mit versehentlich zu langen Alert- oder Button-Texten. Die fertige Box läßt sich zusätzlich anzeigen. Ist sie zur Zufriedenheit fertiggestellt, kann sie abgespeichert werden. Sie wird in einen entsprechenden Aufruf von *Do_Alert* übersetzt, da es sich dabei um eine Funktion handelt, muß noch eine Integer-Variable deklariert werden, um die Buttons auswerten zu können.

Hat man die Möglichkeit der Dialoge gewählt, besteht Auswahl einen Dialog zu erzeugen, vorhandene Dialoge zu verändern, zu laden und zu sichern.

Dialoge erzeugen

Begonnen wird die Dialogerstellung mit der Eingabe der Koordinaten, zur Plazierung des Dialoges. Standardmäßig ist hier jeweils eine Null vorgegeben, was eine Zentrierung des Dialoges bewirkt. Es folgen Eingaben über die Breite und Höhe des Objektes. Eine Breite von 50 Zeichen und eine Höhe von 15 ist auch hier vorgegeben. Das bietet einige Vorteile, falls die genaue Größe noch nicht bekannt sein sollte. Nun kann der noch leere Dialog mit Objekten gefüllt werden. Wie in Bild 2 zu sehen ist, können wie in einem Resource Construction Set Objekte angewählt werden. Ist man sich über das Aussehen eines solchen Objektes nicht im klaren, hilft ein Druck auf die H-Tasten. Hierdurch wird das Laden einiger Hilfsdialoge bewirkt, die die Objekte als Beispiel anzeigen. Nach dem Anwählen eines solchen Ob-

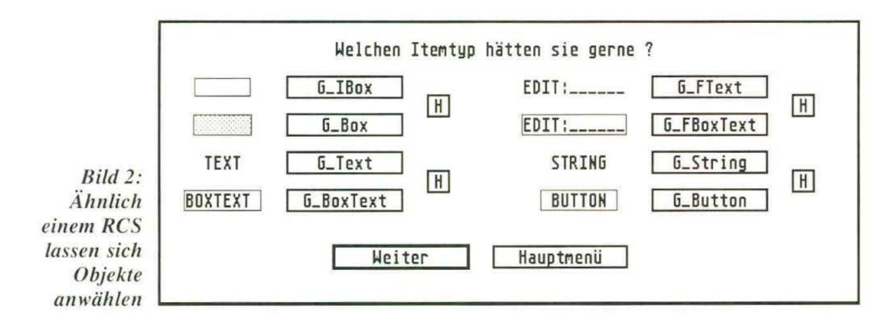


Bild 2: Ähnlich einem RCS lassen sich Objekte anwählen

jektes werden in dem Dialog in Bild 3 zum einen die Koordinaten des Objektes angegeben, zum anderen lassen sich die Flaggs, Farben und Füllmuster komfortabel einstellen. Auch hier erleichtern Hilfsdialoge die Auswahl. Ein so fertiggestelltes Objekt kann zusätzlich noch betrachtet werden. Verläßt man diesen Dialog durch "OK", wird der in Arbeit befindliche Dialog gezeichnet.

Änderungen sind somit sofort möglich und es entfällt das oft sehr lästige permanente Compilieren und Linken eines Dialoges.

Bild 3: Die weitere Definition der Objekte



Bild 4: Beispieldialog zum Listing

Zwar bietet Quick Dialog nicht den hohen Komfort eines Resource Construction Sets, nach kurzer Einarbeitungszeit ist aber eine einfache und schnelle Dialogherstellung kein Problem. Ist der Dialog fertig, kann er als RSC-File abgespeichert werden. Noch interessanter ist natürlich die Möglichkeit, den Dialog direkt in Pascal-Sourcecode verwandeln zu lassen, der direkt eingebunden werden kann. Bild 4 zeigt einen Demonstrationsdialog und sein Pascal-Äquivalent in Listing 1.

Das größte Schmankerl zum Schluß!

Meiner Meinung nach die schönste Fähigkeit von Quick Dialog ist die Möglichkeit, auch mit Construction Sets gefertigte Dialoge in Sourcecode umzuwandeln. Hier bietet sich der maximale Komfort solcher Programme inklusive einer leichten Einbindung in Pascal an. Das ist besonders dann möglich, wenn Quick Dialog als Accessory implementiert wurde und damit direkt jederzeit aus Programmen heraus umgewandelt werden kann.

Das Accessory bietet dazu noch die Möglichkeit, den Dialog-Sourcecode über die Tastatur direkt auszugeben, d.h. es kann in ein gerade im Editor befindliches Programm geschrieben werden.

Wie von einer unsichtbaren Sekretärin geschrieben, erscheinen die Buchstaben an der aktuellen Cursor-Position.

Einschränkungen und Fehler

Einige Einschränkungen müssen leider in Kauf genommen werden. So können Objekte in Dialogen nur im Zeichenrastermaß ausgegeben werden, was jedoch auch an den Pascal-Befehlen liegt, die hier nur Integerwerte zulassen. Eine weitere Einschränkung besteht darin, daß weder Icons noch Grafik möglich sind. Besonders bei den Icons ist das schade, da deren Parameter bei ST-Pascal definiert sind und sie sich zunehmender Beliebtheit erfreuen. Gravierende Fehler konnten von mir nicht entdeckt werden. Manchmal kann es vorkommen, daß bei der Definition der Objektnummernvariablen hinter der letzten die Declaration vom Typ Integer fehlt. Auch bei sehr langen Texten, edierbaren Objekten und Buttons mit sehr viel Text kann es schon mal geschehen, daß der letzte Parameter bei der Objektdefinition verstümmelt wird oder eine Klammer fehlt. Das tritt aber nur bei der Direktausgabe über die Tastatur auf, beim Abspeichern konnte ich dies nicht feststellen.

Fazit

Die oben genannten kleinen Fehler sind unerheblich, da die positiven Funktionen bei weitem überwiegen. Das Programm stellt eine maßgebliche Erleichterung im täglichen Umgang mit ST-Pascal+ dar. Neben den bereits angesprochenen Vorteilen, ist vor allem die immense Zeitersparnis hervorzuheben. Für den Preis von 49.- DM bekommt man ein Hilfsprogramm, das bei keinem Pascal-Programmierer fehlen sollte. Einzig eine Erweiterung für Icons sollte diesem Programm noch beigefügt werden.

Rainer Esser

Bezugsadresse:

CCD
Burgstr.9
6228 Eltville
Tel.06123/1638

```
 1:  FUNCTION test : Dialog_ptr;
 2:  VAR
 3:    Dialog: Dialog_ptr;
 4:    Item1,Item2,Item3,Item4,Item5,Item6,Item7,Item8,Item9:INTEGER;
 5:
 6:  PROCEDURE LEGE_ITEM_UM(DIALOG: Dialog_ptr;ITEM,VATER:INTEGER);
 7:
 8:  TYPE AES_INT_IN=Array[ 0..15]  of INTEGER;
 9:       AES_INT_OUT=Array [0..45] of INTEGER;
10:       AES_ADDR_IN=Array [0..1]  of DIALOG_PTR;
11:       AES_ADDR_OUT=Array[0..0]  of DIALOG_PTR;
12:
13:  VAR INTIN  :AES_INT_IN;
14:      INTOUT :AES_INT_OUT;
15:      ADDRIN :AES_ADDR_IN;
16:      ADDROUT:AES_ADDR_OUT;
17:
18:
19:  PROCEDURE AES_CALL( OP:INTEGER;
20:                  VAR INT_IN :AES_INT_IN ; VAR INT_OUT :AES_INT_OUT;
21:                  VAR ADDR_IN:AES_ADDR_IN; VAR ADDR_OUT:AES_ADDR_OUT);
22:  EXTERNAL;
23:
24:  BEGIN
25:    Intin[0]:=ITEM;
26:    ADDRIN[0]:=DIALOG;
27:    AES_CALL( 41,INTIN,INTOUT,ADDRIN,ADDROUT);
28:    INTIN[0]:=VATER;
29:    INTIN[1]:=ITEM;
30:    ADDRIN[0]:=DIALOG;
31:    AES_CALL( 40,Intin,INTout, addrin,Addrout);
32:  END;
33:
34:  BEGIN
35:    Dialog:= New_Dialog (9,6,6,67,14);
36:    Dialog^[0].ob_Spec.Color:=4417;
37:    Item1:= Add_Ditem (Dialog,G_BoxText,0,1,1,65,2,-1,4209);
38:    Set_Dtext (Dialog, Item1,'Demonstrationsdialog',3,TE_Center);
39:    Item2:= Add_Ditem (Dialog,G_Box,0,1,6,65,4,255,4385);
40:    Item3:= Add_Ditem (Dialog,G_FBoxText,8,1,1,63,2,-1,4480);
41:    Set_Dedit (Dialog, Item3,
42:    'Ihre Meinung zu Quick Dialog : _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _',
       'XXXXXXXXXXXXXXXXXXXXXXXX','',3,TE_Center);
43:    LEGE_ITEM_UM(DIALOG,ITEM3,ITEM2);
44:    Item4:= Add_Ditem (Dialog,G_Button,5,25,11,20,2,0,0);
45:    Set_Dtext (Dialog, Item4,'Finde ich ganz gut', System_Font, TE_Left);
46:    Obj_SetState (Dialog, Item4,48,FALSE);
47:    Item5:= Add_Ditem (Dialog,G_Button,5,47,11,18,2,0,0);
48:    Set_Dtext (Dialog, Item5,'Nicht gut', System_Font, TE_Left);
49:    Obj_SetState (Dialog, Item5,56,FALSE);
50:    Item6:= Add_Ditem (Dialog,G_Button,7,1,11,22,2,0,0);
51:    Set_Dtext (Dialog, Item6,'Finde ich sehr gut', System_Font, TE_Left);
52:    Obj_SetState (Dialog, Item6,48,FALSE);
53:    Item7:= Add_Ditem (Dialog,G_Box,0,1,3,65,2,255,4465);
54:    Item8:= Add_Ditem (Dialog,G_Text,0,0,0,64,1,-1,4096);
55:    Set_Dtext (Dialog, Item8,
56:    'Versuchsdialog zur Übersetzung von RSC-Files in Sourcecode',5,TE_Center);
57:    LEGE_ITEM_UM(DIALOG,ITEM8,ITEM7);
58:    Item9:= Add_Ditem (Dialog,G_Text,32,1,1,64,1,-1,4096);
59:    Set_Dtext (Dialog, Item9,'für ST Pascal + ',5,TE_Center);
60:    LEGE_ITEM_UM(DIALOG,ITEM9,ITEM7);
61:    CENTER_DIALOG(DIALOG);
62:    test:=DIALOG;
63:  END;
```

Demonstrationsdialog
Versuchsdialog zur Übersetzung von RSC-Files in Sourcecode
für ST Pascal +
Ihre Meinung zu Quick Dialog : ________________________
Finde ich sehr gut | Finde ich ganz gut | Nicht gut





Wirkt nachhaltig gegen chronischen Ärger mit der Buchhaltung.

Wirkstoffe: 100.000e wohldosierter Bytes

Anwendungsgebiete:
Problemlose Einnahme-Überschuß-Rechnung (fibuMAN e + m) und Finanzbuchhaltung nach dem neuesten Bilanzrichtliniengesetz (fibuMAN f + m)

Nebenwirkungen:
exzellente Verträglichkeit mit:
fibuSTAT - graphische Betriebsanalyse
faktuMAN - modulares Business-System

Gegenanzeigen:
Verschwendungssucht, akute Aversionen gegen einfache und übersichtliche Buchhaltung

fibuMAN Programme gibt es schon ab DM 398,-*
* unverbindliche Preisempfehlung Atari ST, Preise für fibuMAN MS-DOS® und Apple Macintosh® auf Anfrage

Testsieger in DATA WELT, 6/89
4 MS-DOS® Buchführungsprogramme im Prüfstand; davon 3 mit 8.23, 8.25, 8.65 Punkten (max. 10) fibuMAN mit der höchsten Punktzahl des Tests 9.35

fibuMAN begeistert Anwender wie Fachpresse! Nachzulesen in: ct 4/88, DATA WELT 3/88, 6/88, 5/89, 6/89, ST-COMPUTER 12/87, 12/88, ST-MAGAZIN 4/88, 10/88, ATARI SPECIAL 1/89, ATARI MAGAZIN 8/88, ST-PRAXIS 5/89, ST-VISION 3/89, PC-PLUS 5/89

NEU
1ST fibuMAN
Die Einsteiger-Buchführung
DM 148,-*

novoPLAN SOFTWARE GMBH
Hardtstraße 21 · 4784 Rüthen 3
Tel. (02952) 8080 + (0161) 2215791
Telefax (02952) 3236

Senden Sie mir für fibuMAN ○ INFO ○ Demo mit Handbuch
ich arbeite mit dem System ○ MS-DOS ○ ATARI ○ Macintosh
Mein Name:
in Firma:
Straße/Nr.:
PLZ/Ort.:
Demo mit Handbuch DM 65,-



Low-Cost-Schreiben

Textverarbeitungsystem auf PD-Basis

Natürlich kann man auf käufliche Textverarbeitungsprogramme zurückgreifen. Mit etwas Phantasie ist es aber auch möglich, sich mit Hilfe der ST Public Domain-Sammlung ein den eigenen Ansprüchen angepaßtes Textsystem zusammenzustricken. Die Zutaten für diesen Basteltip sind: ASCII-Edit 1.0 (PD 164), FontEdit 2.0 (PD 217) und das Textverarbeitungsprogramm Minitext 2.1 (PD 230). Wenn dieses PD-Trio richtig aufeinander abgestimmt wird, tun sich dem elektronischen Autor erstaunliche Möglichkeiten auf.

Im Mittelpunkt steht das Textverarbeitungsprogramm Minitext von Heinrich Möller aus Hildesheim. Als er 1984 mit dem "Computern" anfängt, verzichtet er aus Kostengründen auf den Kauf eines Textverarbeitungsprogrammes. Da er nur ein paar kleine Briefe schreiben will, greift er selbst in die Tasten, um zunächst eine Lösung für den Eigenbedarf zu entwikkeln. Nach einem Wechsel 1987 auf den ATARI ST und auf GFA-BASIC liegt mit Minitext 2.56 (ab Version 2.56 mit Wordwrap, Zeilenlineal und schnellerer Bildschirmausgabe) nun das vorläufige Ergebnis als Public Domain-Programm vor. Mit einer Kapazität von 5000 Zeilen mit jeweils 32767 Anschlägen ist es aber zu mehr als nur zum Briefeschreiben zu gebrauchen. Die 5000 Zeilen entsprechen ungefähr 80 Seiten, so daß sich auch Referate und längere Aufsätze mit diesem Programm schreiben lassen.

Nach dem Programmstart befindet man sich im Editor und kann sofort drauflos schreiben. Dabei wird in einer Kopfzeile ständig die Schreibposition im Text angezeigt sowie der Schreibmodus "Einfügen" oder "Überschreiben". Auch Datum und Uhrzeit werden in der rechten oberen Ecke angegeben. Nach dem ersten Abspeichern des bearbeiteten Textes wird zusätzlich auch der komplette Speicherpfad angezeigt, da es nun möglich ist, den Text mit einem Druck auf die F10-Taste ohne den Umweg über die Dateiauswahlbox abzuspeichern. Da sich der Autor allerdings zum Ziel gemacht hat, ein möglichst leicht zu bedienendes Programm zu erstellen, braucht man natürlich nicht Unmengen von Tastenkombinationen auswendigzulernen. Alle Funktionen lassen sich bequem über Drop-down-Menüs aufrufen, wobei die meisten sich zusätzlich auch über eine Taste aktivieren lassen.

Es wäre Unsinn, hier jede einzelne Funktion detailliert zu beschreiben, so daß hier nur die wichtigsten Einrichtungen erwähnt werden sollen. So ist es von Minitext aus möglich, Dateien zu löschen oder umzubenennen, was man oft bei einigen professionellen Programmen schmerzlich vermißt, wenn Platzmangel auf der Diskette herrscht. Natürlich gehören zu einem guten Textverarbeitungsprogramm auch Blockoperationen wie Laden, Speichern, Löschen, Verschieben, Einfügen und Drucken. Dabei darf der Block übrigens bis zu 1000 Zeilen lang sein. Eine interessante Variante der Blockoperationen stellt die Textbaustein-Option dar: Hier wird der Textblock nicht in den Arbeitsspeicher geladen, sondern der Ladepfad von Diskette in den Text eingefügt. Erst beim Ausdrucken wird der Textbaustein gelesen und eingefügt. Es ist allerdings dabei zu berücksichtigen, daß so eingefügte Texte nicht mehr formatiert werden, was Vor- und Nachteile hat. Die Textbaustein-Option spart Speicherplatz und erhöht - sinnvoll angewendet - auch die Übersicht in längeren Texten. Auf diese Weise kann zum Beispiel auch eine "Warteschlange" definiert werden, bei der völlig verschiedene Texte hintereinander ausgedruckt werden.

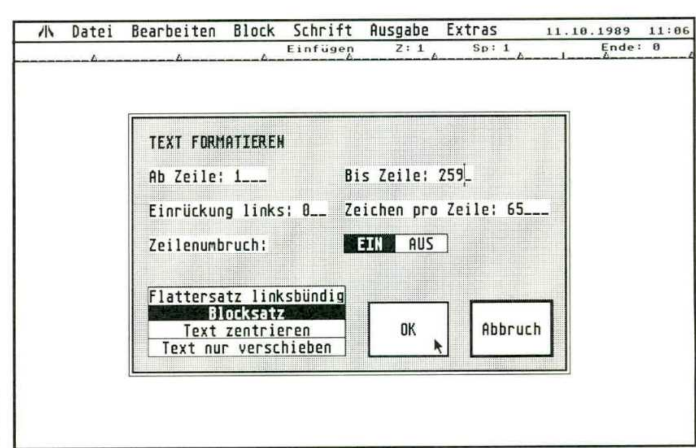


Bild 1: Die Formatieroption von Minitext

Wie zu Beginn angedeutet, bietet der Editor, der übrigens sehr schnell ist,

selbstverständlich verschiedene Bearbeitungshilfen. Neben der bereits erwähnten Anzeige der Schreibposition kann man per Tastendruck an den Textanfang oder das Ende springen. Auch ist nach Angabe einer Zeilennummer eine bestimmte Textzeile erreichbar. Mit den Cursor-Tasten oder der Maus kann man sich beliebig im Text bewegen. Auch eine "Suchen/Ersetzen"-Funktion zur schnellen Fehlerkorrektur ist vorhanden.

Die Tasten Delete, Backspace und Insert bedürfen keiner Erklärung. In Kombination mit der Control-Taste bieten sie allerdings weitere Möglichkeiten: Ctrl+Delete löscht die aktuelle Zeile, Undo fügt sie wieder ein, Crtl+Insert fügt eine Leerzeile ein. Mit dem Linkspfeil gelangt man direkt zum Zeilenanfang, mit dem Rechtspfeil entsprechend an das Zeilenende. Mit den beiden anderen Tasten kann man seitenweise im Text blättern. Zusätzlich mit der Shift-Taste wird der links von der Schreibmarke stehende Text in die vorhergehende Zeile übernommen, mit dem Rechtspfeil der rechts stehende Text in eine neue Zeile geschoben. Über die Help-Taste kann man sich eine kleine Übersicht dieser Kommandos auf den Bildschirm holen, da sich diese Funktionen der Einfachheit halber nicht über Menüs ansprechen lassen.

Ein großer Vorteil eines Matrixdruckers gegenüber einer Schreibmaschine besteht darin, daß er die Schrifttypen variieren kann: fett, breit, schmal, kursiv und unterstrichen ist mit fast jedem Drucker möglich. Minitext unterstützt mit Hilfe einer Druckeranpassung auch diese Möglichkeiten, wobei aber leider die notwendigen Steuerzeichen (noch) auf dem Bildschirm erscheinen und bei der Formatierung nicht als solche erkannt werden. Besonders im Blocksatz sollte man daher diese Attribute erst ganz zum Schluß einfügen. Wer sich übrigens schon öfter über READ.ME-Dateien im Wordplus-Format (*.DOC) geärgert hat, kann sie mit Minitext lesbar ausdrucken, wobei die Steuerzeichen entsprechend umgesetzt bzw. gelöscht werden.

Nachdem nun das Stichwort "Formatierung" gefallen ist, sei kurz auf die Formatier-Option bei Minitext eingegangen. Zu unterscheiden ist dabei nämlich ein Menü, das sich auf die Darstellung am Bildschirm bezieht, und eines, das den Drucker ansteuert. Beim ersteren läßt sich die Textbreite sowie die Ausrichtung (Flattersatz, Blocksatz oder zentriert) einstellen. Zwar kann man auch einen linken Rand setzen, doch dies sollte man aus Speicherplatzgründen erst im Druck-Menü machen. Dort wird dann der Rand nicht über Leerzeichen erzeugt, sondern per Befehl gesetzt. Außerdem wird festgelegt, wie viele Zeilen eine Seite hat und in welcher Schriftart bzw. Druckqualität der Text ausgedruckt werden soll. Der mitgelieferte Druckertreiber sollte für die ersten Versuche ausreichend sein. Wenn man mit seinem Drucker und Minitext schon etwas vertraut ist, kann man sich leicht eine eigene Anpassung schreiben, da Kommentare jede einzelne Zeile erläutern.

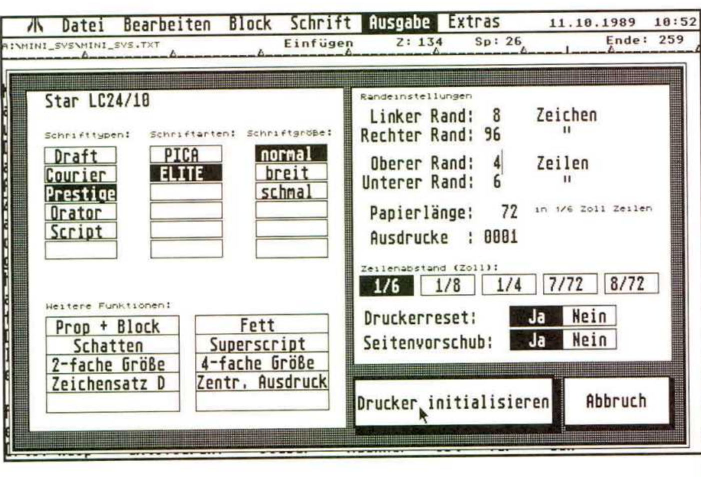


Bild 2: Umfangreiche Druckeransteuerung ist möglich.

Bild 4: Die Oberfläche von FontEdit

Bild 3: Der eingebaute UPN-Rechner

Fast schon zum guten Ton eines Textverarbeitungsprogrammes gehört ein integrierter Taschenrechner. Bei Minitext läßt er sich über Crtl+Help aktivieren. Dieser Rechner ist für den Normalverbraucher allerdings etwas gewöhnungsbedürftig, da er sich der umgekehrten polnischen Notation (UPN) bedient. Dabei werden zuerst die Zahlen und anschließend die Rechenoperation angegeben. Auf diese Weise werden zwar Klammern überflüssig, man muß aber erst eine Weile üben. Der Vorteil gegenüber den meisten Rechner-Accessories besteht aber darin, daß das angezeigte Ergebnis beim Verlassen des Rechners direkt in den Text übernommen werden kann.

FontEdit

Nun wird es aber Zeit, daß die Rolle der beiden anderen oben erwähnten Programme endlich erläutert wird. Sie passen nämlich sehr gut zum Konzept der individuellen Anpassung. Das Programm FontEdit 2.0 von Dirk Woitha ist zur Anpassung von GEM-Zeichensätzen an eigene

Vorstellungen gedacht. Die Handhabung ist äußerst einfach und bedarf kaum einer Erklärung, zumal hier auf den speziellen Nutzen in der Textverarbeitung eingegangen werden soll. Nach Aufruf des Accessorys wird der aktuelle Zeichensatz in einer Tabelle dargestellt. Das Zeichen, das verändert werden soll, wird einfach angeklickt und erscheint in dem üblichen Rasterfeld, wo man es nach Belieben verändern kann. Anschließend läßt sich der veränderte Zeichensatz abspeichern und im System installieren.

Da mein Drucker - wie viele andere - einerseits auch über die IBM-Blockgrafik verfügt, ich aber überhaupt keine Verwendung für den hebräischen Zeichensatz im ATARI habe, bietet sich eine entsprechende Anpassung doch geradezu an. Dies ist auch gar nicht besonders mühselig, da dem Zeichensatz-Editor (zufällig?) ein entsprechender IBM-Zeichensatz als Beispiel beiliegt. Da dieser Anpassung das Paragraph-Zeichen zum Opfer fällt, sollte man es an der Position 221 (dd) wieder einfügen und die Tauschtabelle des Minitexttreibers entsprechend anpassen (#71 e:dd a:15). Wenn man diesen neuen Zeichensatz in DEFAULT.FNT umbenennt, wird er von dem Accessory beim Systemstart automatisch installiert.

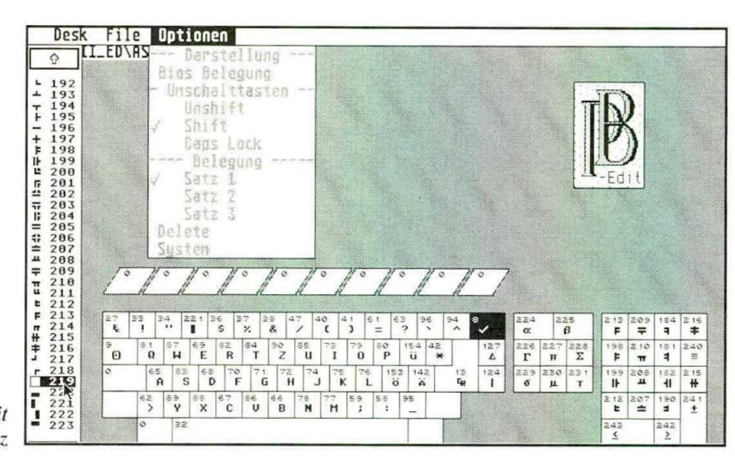


Bild 5: ASCII-Edit im Einsatz

Numero 3: ASCII-Edit

Da in Minitext über eine spezielle Funktion jedes ASCII-Zeichen in den Text eingefügt werden kann, kann man auch direkt mit Tabellen und griechischen Buchstaben seinen Text auflockern. Doch obwohl die "ASCII einfügen"-Funktion für einzelne Zeichen recht gut zu gebrauchen ist, ist das Erstellen von Tabellen etc. auf diese Weise doch nicht ganz einfach. Da die ATARI-Tastatur über mehr Tasten als für eine Schreibmaschinentastatur unbedingt nötig verfügt, bietet es sich doch geradezu an, den Ziffernblock für den IBM-Zeichensatz zu mißbrauchen. Hier kommt nun das oben erwähnte ASCII-Edit 1.0 von Peter Berghammer zum Zuge. Mit diesem Programm ist es nämlich möglich, die gesamte Tastenbelegung (auch die Funktions- und Cursor-Tasten) eigenen Erfordernissen anzupassen. Dazu brauchen nur eine Taste sowie das gewünschte Zeichen angeklickt zu werden. Es können jeweils die Ebenen "normal", "Shift" und "Caps Lock" unterschieden werden.

Im Kombination mit Minitext empfiehlt es sich natürlich, die Standardtastatur sowie die Funktionstasten möglichst unberührt zu lassen, um unangenehme Überraschungen zu vermeiden. Dafür kann man sich auf dem Ziffernblock frei austoben. Durch Dreifachbelegung der 18 Tasten kann man immerhin 54 neue Zeichen auf die Tasten zaubern, was meiner Meinung nach völlig ausreicht. Durch eine sinnvolle Anordnung auf den Tasten kann man problemlos auf die Zeichen zurückgreifen. Mit einem speziellen Hilfsprogramm wird diese neue Tastenbelegung beim Systemstart automatisch eingerichtet.

man seine Texte gar nicht ausdrucken, sondern nur in einem Programm auf dem Bildschirm darstellen will, kann man natürlich auch im Font-Editor seiner Phantasie freien Lauf lassen und die kuriosesten Figuren auf den Monitor zaubern.

(	)	/	*
7	8	9	-
4	5	6	+
1	2	3	
0		.	

┌	─	┐	⌠
│	┬	│	⌡
├	┴	┤	┼
└	─	┘	
°		·	√

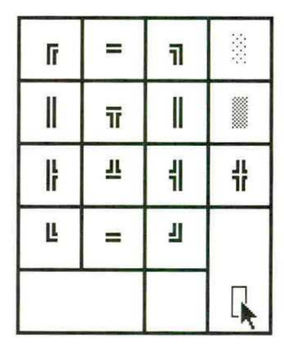


Bild 6: Mögliche Alternativ-Belegungen des Zehnerblocks

Obwohl ASCII-Edit kein Accessory ist, kann man die Tastaturbelegung auch vom Textprogramm aus jederzeit ändern. Minitext bietet nämlich die Möglichkeit, zwischendurch ein anderes Programm aufzurufen. Wird dieses "externe" Programm beendet, landet man nicht wie gewohnt im Desktop, sondern wieder im Text-Editor. Da allerdings immer die Gefahr besteht, daß das zweite Programm einmal abstürzt, sollte der gerade bearbeitete Text vorher sicherheitshalber abgespeichert werden, um sich unnötigen Ärger zu ersparen. - Man braucht ja nur kurz F10 zu drücken.

Natürlich bieten sich noch andere Anwendungsmöglichkeiten für diese beiden Programme. Wer öfters französische Texte schreiben muß, kann sich auch die entsprechenden Sonderzeichen (Ç, é, â, à, ê, ë, è, ï) auf den Ziffernblock legen. Der Mathematiker nimmt "seine" Zeichen, sofern der Drucker sie zu Papier bringen kann (≤, ≥, ±, √, ½, ¼, ∂, ß, ¥, π). Wenn

Meines Erachtens stellt das hier vorgestellte PD-Gespann eine sehr gute Lösung für den angehenden Textverarbeiter dar. Das empfehlenswerte Textverarbeitungsprogramm Minitext wird durch die Programme ASCII-Edit und Font-Editor sinnvoll ergänzt. Selbstverständlich sind noch andere Kombinationen möglich, die hier nicht alle aufgezeigt werden können (wie zum Beispiel ein Ausdruck der erstellten Texte mit Fontmaster von PD 40). Natürlich können der Font-Editor und das Tastaturbelegungs-Tool auch für andere Textsysteme benutzt werden. Wordplus freut sich sicherlich über diese neuen Features.

Thorsten Luhm

Bezugsadresse:
Maxon Computer
Schwalbacher Str. 52
6236 Eschborn
Tel.: 06196/481811

That's Pixel

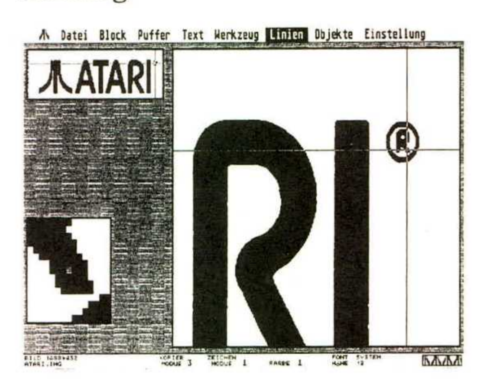
148,- DM

Nur ein Malprogramm? Das Grafiktool mit Konzept!

Etwas Statistik: Es gibt etwa 200 Malprogramme für den ST. Davon sind ca. 25 in der Lage, Grafiken, größer als der Bildschirm, zu bearbeiten. Davon erlaubt wieder nur ein Teil der Programme einen bequemen Umgang mit dem Bild.
Kein weiterer Kommentar!

- Bildgröße nur durch Speicher begrenzt
- Bildmanipulation leicht
- Montage mehrerer Bilder, auch aus Fremdprogrammen, zu einem großen Bild
- bedienbar mit Maus- und Tastatur
- viele praktische Hilfs- und Zeichenfunktionen
- läuft unter SM124 oder auch mit MegaScreen

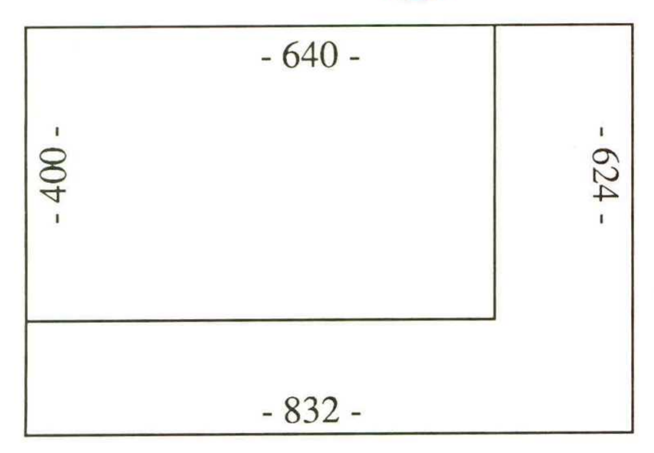
Gleichzeitig und daher übersichtlich bearbeitet man Ganzseitendarstellung, Lupe und 1:1-Darstellung



DM 148,-
Versandkosten DM 6,-
DM 154,-

MegaScreen

298,- DM



Die preiswerte Grafikkarte für den Mega ST
erweitert niedrige, mittlere und hohe Auflösung,
z.B.: 640x350 in mittlerer Auflösung,
832x624 in hoher Auflösung.
Die Auflösung kann vom Desktop aus geändert werden. Volle Kompatibilität auch bei Programmen, die nur mit der original Auflösung arbeiten, da diese eingestellt werden kann.

DM 298,-
Versandkosten DM 6,-
DM 304,-

– Alle genannten Preise unverbindliche Preisempfehlungen –

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Hotline
Info + Betreuung
COMPO SOFTWARE GMBH
Ritzstr. 13
5540 Prüm
Tel.: 0 65 51 / 62 66

BESTELL-COUPON

Bitte senden Sie mir: ______ That's Pixel à DM 154,-
______ MegaScreen à DM 304,-

Name, Vorname ______________________

Straße, Hausnr. ______________________

PLZ, Ort ______________________

Oder benutzen Sie die in ST-Computer eingeheftete Bestellkarte

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

In der Schweiz:
an Data Trade AG
Landstr. 1
CH - 5415 Rieden-Baden

In Österreich:
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

Manhatten ST

Hat man 'en ST?

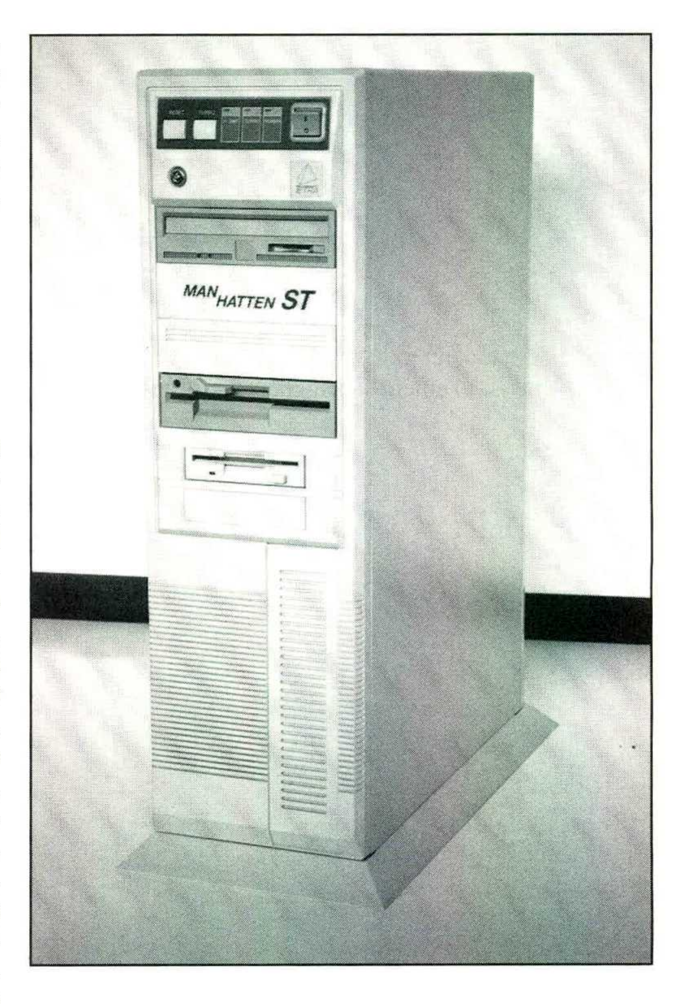
Man hat 'en ST! Sonst würde man wahrscheinlich kaum diese Zeilen lesen. Wenn man allerdings den "Manhatten ST" vor sich stehen sieht, glaubt man nicht mehr, daß man einen ST hat. Manhatten ST ist ein High-End-Tower-ST, gegen den ein Mega ST wie ein ZX 81 wirkt. Viel interessanter ist aber, was im Tower seinen Dienst verrichtet...

Ich spanne Sie noch ein wenig auf die Folter, indem ich Ihnen zuerst aufzähle, was sich alles auf den Außenseiten des turmförmig aufgeschichteten ST befindet. Auf der Vorderseite befinden sich zunächst drei Schalter. Der erste enthält die Aufschift "I" und "0". Damit dürften wir zweifellos den Ein-/Ausschalter gefunden haben. Auf der linken Seite befinden sich zwei Schalter mit der Aufschrift "Reset" und "Turbo". Bei "Turbo" kommt mir sofort der Gedanke, daß es sich hier vielleicht um einen verkappten Plantron- oder Acer-Rechner handelt. Die wahre Erklärung dazu folgt weiter unten. Die Bezeichnung Reset dürfte klar sein, das ist der Knopf, der gedrückt werden muß, wenn sich der ST aus dem Leben bombt. Zwischen den Schaltern befinden sich Leuchtdioden, die mit "H-DISK", "TURBO" und "POWER" beschriftet sind. H-DISK leuchtet immer dann auf, wenn ein Zugriff auf die Festplatte stattfindet, die Turbo-Lechtdiode immer dann, wenn man den entsprechenden Knopf drückt. Power wiederum dürfte klar sein, es ist die Leuchtdiode, die zuverlässig die Betriebsbereitschaft des Geräts anzeigt.

Unter der Knopfreihe befindet sich ein Schlüsselschalter, mit dem sich den Tower wirkungsvoll vor fremden Zugriffen schützen läßt. Daneben findet man das Zeichen des Herstellers: "Tetra" - also nicht, wie vermutet, Acer oder Plantron. Unter diesen Elementen enthält das Gehäuse bekannte Frontplatten: Die folgende sieht wie eine Wechselplatte aus, darunter vermutet man die Deckplatte einer Festplatte, dann folgend ein $5^1/_4$"- und dann ein $3^1/_2$"-Laufwerk.

...und von hinten

Auch die Rückseite bietet einen interessanten Anblick, sind hier doch Anschlüsse zu finden, bei denen man ebenfalls wieder auf Acer oder Plantron schließen würde. Von oben nach unten gibt es hier einen Umschalter für 220 oder 110 Volt, einen überdimensionalen Lüfter samt Metallgitter, einen Stromein- und -ausgang, eine RJ11-Buchse, wie sie für amerikanische Telefone und ATARI-Tastaturen benutzt wird, ein Trimmpoti, eine Schnittstelle mit der Aufschrift "Modem RS232", eine mit "Printer parallel", ganz unten befinden sich schließlich zwei Buchsen mit der Aufschrift "Color Monitor" und "Mono Monitor". Wo man allerdings eine 13polige ATARI-Buchse für den SM124 vermuten würde, existieren stattdessen zwei 9polige SUB-D-Buchsen, wie sie bei IBM-Clones benutzt werden. Nicht zu finden sind ein DMA-Anschluß, MIDI-Buchsen und Anschlußbuchsen für einen SM124. Einen ROM-Port sucht man ebenfalls vergebens (MAC-Emulator oder ROM-Bank, ade!).

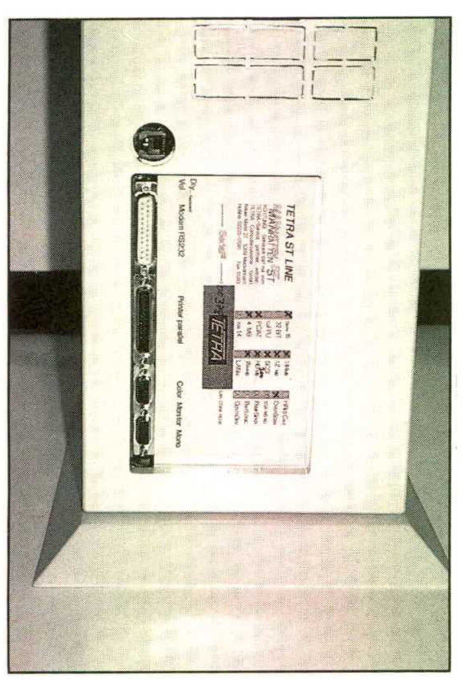


Auch der Monitor ist absolut ATARI-untypisch. Er trägt die Aufschrift "Dual-Display". Ein äquivalentes Gerät habe ich bereits mit der Aufschrift "IBM" (Insufficent Batch Machines) in der rechten oberen Ecke gesehen. In unserem Fall befindet sich hier eine Einbuchtung an der Stelle, dafür steht links unten wieder "Tetra" auf einem Aufkleber. Die Tastatur erinnert stark an die Mega ST-Tastatur. Allerdings fehlt das ATARI-Zeichen in der rechten oberen Ecke, es ist überklebt worden mit einem weiteren Tetra-Aufkleber. Auf der mitgelieferten Maus ist das

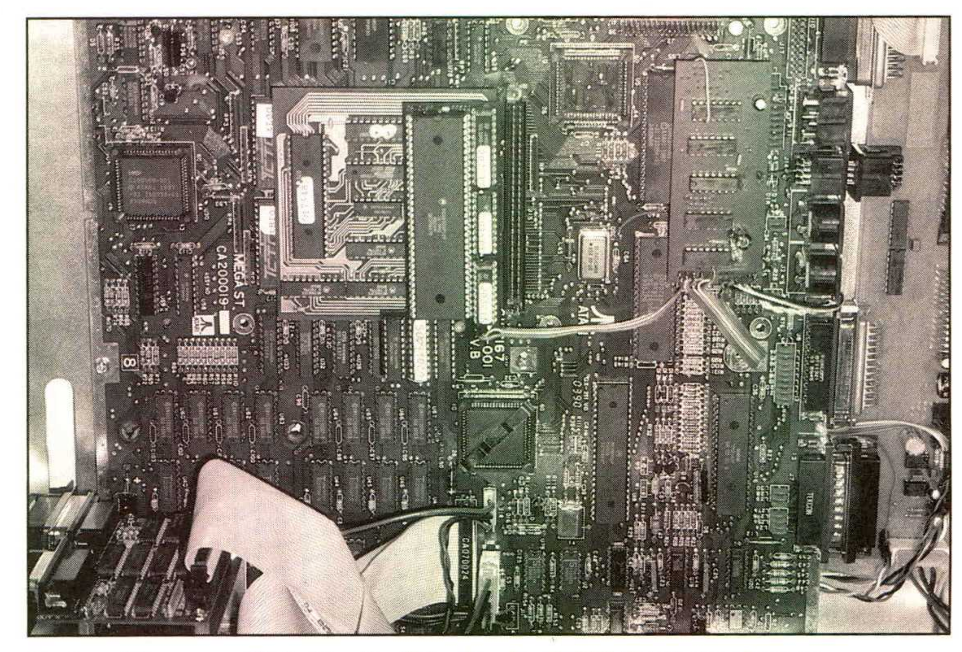
Raketenzeichen von ATARI jedoch noch zu sehen.

Klopf, klopf!

Der Volksmund meint trefflich: Klopfet, so wird Euch geöffnet. Da auf mein eindringlich lautes Klopfen an der Außenseite des Towers niemand öffnet, greife ich zum Schraubendreher, um dem Gerät damit unverzüglich und gnadenlos zu Leibe zu rücken ("O Doktor Martin, wird er die Operation überleben?"). Nachdem die Ummantelung des Geräts abgeschält ist, werfe ich einen kurzen Blick in das Bauwerk. Meine Kinnlade fällt zu Boden und schlägt hart auf, denn die Innereien dieses Geräts können eigentlich nicht zusammen funktionieren. Den größten Teil des Geräts nimmt eine Mega ST 4-Platine ein, die hochkant auf der rechten Seite aufgestellt ist. Hinter ihr befindet sich eine weitere, die normalerweise nicht in einem ST 4 zu finden ist. Auf ihr sind die Anschlußbuchsen angebracht, die an der Hinterseite des Towers zu finden sind. Es handelt sich also lediglich um einen "künstlichen Ausgang" des Mega, sprich: um eine Verlängerung der Platine. Die nächstgrößere Innerei trägt die Aufschrift "220 W" und ist in einem silbrigglänzenden Kasten verstaut. Wir haben zweifellos das Netzteil entdeckt, das mit 220 W vielleicht sogar überdimensioniert ist. Trotzdem ist es längst nicht so warm wie mein CD-Player, nein, sogar angenehm kühl. Damit ist die rechte Seite des Geräts fast erschöpft, lediglich eine Leuchtdiode, die beim Einschalten blinkt, ist zu sehen.



Ein Blick auf die Schnittstellen des Manhatten ST. Auffällig sind die beiden Monitoranschlüsse.



Das Innenleben unseres Testgerätes. Deutlich erkennbar sind PC-Speed mit aufgesetzter 16 MHz-Erweiterung und die blaue Platine für den Anschluß des HD-Laufwerks.

Betrachtet man die Vorderseite, versteht man auch, wieso die höllische Elektronik in einem Tower untergebracht ist. Hier stapelt sich die Crème de la Crème in einer gefährlichen Höhe von fast 30cm auf. Man könnte sagen: Jeder Zentimeter dieses Stapels ist DM 300,- wert. Was von außen wie eine Wechselplatte aussah, entpuppt sich bei genauerer Betrachtung - wie könnte es anders sein - als Wechselplatte von Syquest (SQ555). Direkt darunter befindet sich ein Gerät mit der Aufschrift "Maxtor LXT-200S". Ein kurzer Blick in mein Pfadfinderhandbuch der Fieselschweiflinge verrät mir nichts genaues, trotzdem handelt es sich hier eindeutig um eine 200 MB SCSI-Festplatte mit einer Zugriffszeit von knapp 22 ms - und das bei einer Größe von $3^1/_2$"! Damit dürfte das Ende der Fahnenstange in bezug auf die Schnelligkeit und die Größe erreicht sein. Unter der Luxusplatte befindet sich ein Laufwerk mit der Bezeichnung "Chinon FZ-506". Wieder kann mir mein Allround-Handbuch nicht helfen, obwohl es sich hier augenscheinlich um ein 1,2 MB-Laufwerk im Format $5^1/_4$" handelt. Allerdings verarbeitet es auch problemlos 720k- und 360k-Disketten. Ein Stockwerk tiefer, im Erdgeschoß, meint der Aufkleber eines Laufwerks: "Chinon FD-235HF". Hier handelt es sich eindeutig um ein 1,44 MB-Laufwerk im Format $3^1/_2$". Natürlich können auch hier problemlos 720k- und 360k-Disketten verwendet werden. Damit wären wir fast am Ende, wenn da nicht noch diverse andere "Kleinigkeiten" zu finden wären... Unter dem 1,44er-Laufwerk, also im Kellergeschoß, existiert ein Lautsprecher. Es ist also kein Monitor mit Lautsprecher oder gar eine Stereoanlage notwendig, um das Piepsen der Tastatur zu hören, sondern der Tower gibt in guter IBM-Manier die Piepser direkt aus. In eben dieser Manier ist auch gelöst, daß man die Lautstärke des Piepsers nicht ohne weiteres regeln kann - dazu muß man dem Gerät mit einem Schraubenzieher zu Leibe rücken und das Trimmpoti auf der Rückseite verstellen. Da dieses jedoch gut erreichbar ist und meistens sowieso immer die gleiche Lautstärke eingestellt bleiben soll, ist diese Lösung vielleicht gar nicht so schlecht.

Sprinter

Damit die Festplatte auch richtig losspurten kann, ist sie an einen ICD-Advantage angeschlossen - das ist der neue SCSI-Host-Adapter von ICD, womit der Tower eine Übertragungsrate von weit über 750 kB/Sekunde erreicht! Nun werden Sie sich fragen, wie man mit einem einzigen Schalter die Festplatte(n) und den Rechner einschalten kann? Richtig, da war doch noch eine Leuchtdiode, die beim Einschalten wild blinkt! Ihre Vermutung, daß es sich dabei um eine Einschaltverzögerung für den Rechner handelt, ist korrekt. Bei der Maxtor-Platte ist sie zwar nicht vonnöten, da dieses gerät so schnell bootet, daß man sich zwischendurch nicht mal eine Zigarette anzünden kann; die Wechselplatte ist allerdings etwas langsamer und benötigt die Verzögerung, damit der schnelle Rechner noch mitkommt. Auch die Dauer der Einschaltverzögerung läßt sich mit einem Trimmpoti stufenlos einstellen und ist damit für spätere Erweiterungen bestens ausgerüstet.

Nicht nur die Festplatte spurtet, sondern auch der ST selbst. Sie erinnern sich an den "Turbo"-Schalter auf der Vordersei-


VERSION: 3.7= Dreidimensionales CAD-System mit
File-Name = DREIZYMO.C3D Datum: 06.03.1990 Uhrzeit: 05:26:16
EIZO FLEXSCAN

Die Ergebnisse eines Mega ST mit Turbo ST

Manhatten ST mit Turbo ST im Normal-Modus

Manhatten ST mit Turbo ST im Turbo-Modus

te? Formel I-ähnlich läßt sich damit ein Nachbrenner einschalten, der den ST zum Hecheln bringt. Und richtig, auf der Hauptplatine findet sich bei genauerer Betrachtung eine 16 MHz-Erweiterung mit 16 kB Cache-Speicher. Dadurch wird die Geschwindigkeit des Rechners nahezu verdoppelt, also von knapp 8 MHz auf etwas weniger als 16 MHz aufgebohrt.

Die Erweiterung wird allerdings umlagert von einer anderen Platine, die ich irgendwo schon einmal gesehen habe. Kurze Nachforschungen ergaben, daß es sich hierbei um einen PC-Speed handelt. Damit läßt sich der Tower auch als IBM-Clone benutzen, d.h. man kann sich auch einen Clone-Aufkleber auf das Gehäuse kleben - es dürfte nicht auffallen, daß es sich bei dem Gerät um einen verkappten ST handelt.

Ist hier noch frei?

"...Nein, ich halte den Platz für meinen Freund frei". Diese typische Kinosituation werden Sie mit dem Manhatten ST nicht erleben: Im Gehäuse ist noch massig Platz für alle möglichen Erweiterungen, auch das Netzteil kann noch stark belastet werden, ohne daß es sich in eine Qualmwolke auflöst. Der mitgelieferte Monitor ist ein handelsüblicher Graustufen-Multi-Sync, der alle Auflösungen des ST problemlos und ohne Summ- und Piepgeräusche darstellt.

Bei unserem Gerät fehlten einige Buchsen: Midi, ROM-Port, DMA und Monitorbuchse. Auf Wunsch werden allerdings alle benötigten Buchsen nach außen gelegt (etwa für einen Laser-Drucker). Etwas problematisch wird dagegen schon der Anschluß einer Erweiterung, die auf den ROM-Port aufgesteckt wird - man müßte sie theoretisch im Fußboden versenken oder die aufgesteckte Platine abknicken, was für eine Platine naturgemäß nicht sehr bekömmlich ist. Hier hilft nur ein kurzes Verlängerungskabel, was viele Erweiterungen wegen der guten Pufferung des Busses (!) aber gar nicht mögen. Einzige Lösung: Die Platine muß um 90 oder 180° gedreht und wieder befestigt werden.

Noch mehr?

Wünschen Sie noch mehr Erweiterungen? Tetra baut Ihnen alles ein, was nicht niet- und nagelfest ist. So können Sie Ihren Wunsch-ST zusammenstellen lassen und DOS-User damit schocken - selbst ein 33 MHz-386er dürfte gegen dieses Gerät ziemlich schwach in den Knochen sein. Vielleicht suchen Sie aber auch eine Grafik-Workstation? Dann können Sie das Gerät mit einer eingebauten MGE-Grafikkarte erwerben - für einen entsprechenden Aufpreis, versteht sich. Und: natürlich auch im ansprechenden Gehäuse versenkt.

Nun wissen Sie, welche Erweiterungen sich im Manhatten befinden. Wundersamerweise funktionieren Sie auch alle zusammen! Einen adäquaten Geschwindigkeitstest für einen ATARI ST zu finden, ist recht schwierig. Wir haben Quick ST (ST-PD 275) benutzt, um Vergleichswerte zu einem "normalen" ST zu erhalten - die Werte finden Sie in den einzelnen Abbildungen.

Außergewöhnliche Wünsche kosten außergewöhnlich viel Geld. Leider verhält es sich auch bei unserem Tetra ST-Testgerät so, denn für das komplette Gerät, wie es bei uns stand, müssen knapp DM 12.000,- über die Theke gereicht werden. Der Preis hängt allerdings ganz von der Ausbaustufe ab. Dafür erhalten Sie allerdings auch einen handgemachten ST mit Erweiterungen, die zusammen funktionieren (!), eine installierte Festplatte mit der Software, die Sie sich wünschen, diverse Anleitungen, ein funktionelles Gehäuse sowie einen hübschen Monitor. Vergessen Sie nicht den letzten Vorteil: Nur ein Kabel führt zur Steckdose!

MP

Bezugsquelle:

Tetra Computersysteme GmbH
Neuer Markt 27
5309 Meckenheim
Tel. 02225/17081

Turbo ST 1.8

Der Software-Blitter mit neuem Schliff

Fast jedem ST-Anwender ist mittlerweile Turbo ST, der Software-Blitter der Eschborner Firma Bela Computer, ein Begriff. Hier ist man den Weg gegangen, Geschwindigkeit durch effektivere Programmierung und neue Algorithmen, nicht abr durch Hardware zu erreichen. Gerade TOS des ST bietet ja genügend Ansatzpunkte zum Verbessern.

Durch systematisches Ersetzen von "langsamen" Teilen des GEMDOS, VDI und BIOS durch optimierte eigene Routinen und konsequenter Nutzung der ST-Hardware sorgt das Programm nach wie vor für verblüffende Ergebnisse bei der Ausgabe von Texten und Bildern auf dem Bildschirm. Ebenfalls beschleunigt wird von Turbo ST der Aufbau von Fenstern und Dialogboxen.

Natürlich ersetzt Turbo ST keinen schnelleren Prozessor oder einen Arithmetik-Chip, da wirklich nur die Ausgabe auf den Bildschirm schneller wird. Was Turbo ST wirklich leistet, haben wir anhand einiger Testdurchläufe probiert. Zuerst wurde das PD-Programm Quick Index (PD 275), das teilweise auch Bildschirmoperationen testet, auf Turbo ST losgelassen. Das Ergebnis kann man in der Tabelle bewundern. Angegeben ist immer das prozentuale Verhältnis zwischen einem "normalen" Mega ST mit Blitter und demselben Rechner mit installiertem Turbo ST. Anschließend haben wir noch die Scroll-Geschwindigkeit und das Seitenblättern von Wordplus mit und ohne Turbo ST verglichen.

Versionsvielfalt

Sieht man sich das Inhaltsverzeichnis der Turbo ST-Diskette an, weiß man zunächst überhaupt nicht, was zur Installation alles benötigt wird; denn mittlerweile sind im Lieferumfang des Turbo ST mehrere Versionen enthalten. Ein Blick ins Handbuch schafft sofort Klarheit. Es wird eine Version für Monochrom-, eine für Farb- und sogar eine für einen Moniterm-Großbildschirm (Original-ATARI-Großbildschirm SM 194) geliefert. Durch diese Aufteilung war es möglich, eine höhere Optimierung für die jeweilige Auflösung zu erreichen.

Da Turbo ST früher nur als Accessory geliefert wurde, gab es bei Anwendern, die schon sechs dieser praktischen Helfer in ihrer Menüleiste installiert hatten, Probleme. TOS läßt eben nur diese Anzahl zu. Aus diesem Grund findet man auf der Turbo ST-Diskette auch jeweils eine Version für den Auto-Ordner, so daß kein Platz in der Menüleiste belegt wird. Es bleibt also jedem selbst überlassen, wie er seinen ST auf Touren bringen will. Generell läuft die Accessory-Version schneller als die Auto-Ordner-Version. Mit einem kleinen Trick erreicht man allerdings fast das gleiche Ergebnis. Dazu muß man nur dafür sorgen, daß die Auto-Ordner-Version als letztes Programm des Ordners geladen wird.

Sicherlich kann es vorkommen, daß bestimmte Programme partout nicht mit Turbo ST zusammenarbeiten wollen. Dafür gibt es die Möglichkeit, Turbo ST einfach per Mausklick in einer Alertbox zu deaktivieren und dort auch später wieder einzuschalten. Nach einmonatigem Test ist uns in der Redaktion ein solches Programm allerdings noch nicht begegnet.

Neu in der Version 1.8 ist übrigens auch die Möglichkeit des Ein- und Ausschaltens von Turbo ST aus eigenen Programmen oder Accessories heraus über Message-Pipeline. Auf die gleiche Art und Weise kann man den Status (an/aus) ebenfalls abfragen.

Rundherum kann man sagen, daß Turbo ST auch in der neuen Version 1.8 weiterhin mit an der Spitze der ST-Utilities stehen dürfte. Wer Wert auf einen preisgünstigen schnellen Bildschirmaufbau legt, kommt derzeit nicht an Turbo ST vorbei. Der Preis von DM 89,- ist für den "Software-Blitter" sicherlich nicht zu viel.

HE

Bezugsadresse:

Bela Computer
Unterortstr. 23-25
6236 Eschborn
Tel.: 06196/481944

Quick-Index-Test

	ohne Turbo ST	mit Turbo ST
Bconout-Textausgabe	110%	326%
Cconws-String-Ausgabe	106%	1347%
Bildschirm-Scrollen	132%	139%
GEM-Draw-Test	133%	299%

Wordplus-Test
mit 40 KByte Text

	ohne Turbo ST	mit Turbo ST
Text-Scrollen	2:35:08	1:48:35
Seitenblättern	0:38:25	0:21:02

Galenius

Ein großer Helfer in der Pharmazie



In der Pharmazie bestehen strenge Richtlinien, welche die Erstellung von Rezepturen, Herstellungsanweisungen und Chargenberichten betreffen. So müssen Prüfmuster als maschinengeschriebene und handschriftlich nicht veränderbare Herstellervorschrift existieren. Hier setzt Galenius an, indem es die Erstellung von Formausdrucken ermöglicht, welche den Herstellungsweg des Medikamentes vom Abwiegen bis zur Fertigstellung begleiten und den oben erwähnten Richtlinien entsprechen. Diese enthalten zunächst nur die Sollwerte und keine handschriftlichen Eintragungen. Die Istwerte, Doublechecks und Unterschriften kommen während des Herstellprozesses handschriftlich dazu, wodurch diese zu Chargenberichten werden, die jederzeit mit den gespeicherten Daten verglichen werden können.

Galenius wird in einem festen Schnellhefter im Format DIN A4 ausgeliefert. Dieser enthält eine Diskette mit dem Programm, die vorgefertigten Formularrümpfe und einige Hilfsprogramme für die Ansteuerung des Druckers.
Im Programm befindet sich ein Editor, der die einfachsten Grundbedürfnisse erfüllt, trotzdem ist ein zusätzlicher Editor sehr zu empfehlen. Dieser wird auch von Galenius unterstützt.

Das Handbuch ist ausführlich und übersichtlich geschrieben. Es enthält auch leicht nachvollziehbare Ablaufpläne zur Erstellung des Batch-Reportes und eigener Formulare.

Start mit Hindernissen

Ich wollte natürlich das Programm starten. Leider ergab sich dabei mit meinem Mega ST ein kleines, aber einschneidendes Problem. Es erschien immer die Meldung *Galenius läuft nur mit 640*400 Punkten.* Der Fehler konnte von mir dahingehend verifiziert werden, daß Galenius nur mit dem alten ROM-TOS funktionierte, weil es die Auflösung über Systemvariablen abfragte, die ATARI in neueren TOS-Versionen in andere Speicherbereiche gelegt hat.

Dies führte dazu, die von Softchemie gewährte Qualitätsgarantie in Anspruch zu nehmen. Diese beinhaltet eine kostenlose Fehlerkorrektur für die ersten sechs Monate nach dem Kauf und nach dieser Zeit eine Behebung von Fehlern zum Selbstkostenpreis. Im Handbuch wird gleich ein vorgefertigter Brief mitgeliefert, in dem man den Fehler genau spezifizieren kann. Innerhalb weniger Tage bereits erhielt ich Galenius zurück, und alle von mir gefundenen Fehler sind beseitigt worden. Zwischenzeitlich arbeitet das Programm zur vollen Zufriedenheit. Es ist aber gut zu wissen, daß im Fall des Falles ein guter und schneller Service zu erwarten ist.

Doch nun wenden wir uns dem Ablauf zur Erstellung eines Batch-Reports zu, denn hierzu wird ein Großteil des Programms benötigt.

Der Batch-Report

Der Menüpunkt *Lesen* beeinhaltet den Nur-lese-Zugriff auf alle Dateien des Systems. Diese gliedern sich in Formulare *.F, Rezepturen *.REZ, Ansätze *.REC und Reports *.REP. Weiterhin ist es möglich, den Prozeßordner zu öffnen und sich die mit Standardwerten erstellten Prozeßdaten anzeigen zu lassen. Am Anfang eines Batch-Reportes steht immer die Erstellung der Rezeptur und des Ansatzes für die Herstellung. Dazu wird der Menüpunkt *Planen* angewählt. Das Programm fragt bei Rezeptur- und Ansatzentwicklung nach, ob ein bereits bestehendes File geladen werden soll oder ein neues entwickelt wird. Es erscheint die Eingabemaske in Bild 2. Sie ist für beide Programmpunkte gleich und unterscheidet sich nur in der Mengenangabe bei Rezeptur in mg/Dosis und bei Ansatz in g/Ansatz. Hier ist es auch möglich, Ansätze in kg/Ans. und Stück/Ans. zu machen. Durch Mausklick oder Drücken der Funktionstaste F2 kann eine neue Zeile selektiert werden. Galenius numeriert die Zeilen automatisch durch, so daß nur der Substanzname und die eingesetzte Menge eingegeben werden müssen. Es stehen pro Seite 18 Zeilen zur Verfügung. Die maximale Zeilenzahl liegt bei 38 Zeilen. Eine Fußzeile kann ebenfalls beschriftet werden, und mittels *Mark.LM* ist es möglich, Lösungsmittel, die nicht in die Summe der Feststoffe aufgenommen werden sollen, zu markieren, damit diese bei Anwählen von *Abschluß* bei der Berechnung der Feststoffsumme nicht mehr berücksichtigt werden. Zusätzlich lassen sich Berechnungen mit dem Taschenrechner durch Anwählen von F5 durchführen. Ist man mit dem Ergebnis zufrieden, kann es mit *Speichern* auf Diskette abgespeichert werden.

Lesen	Planen	Prozeß	Berichte	System	Tutorial
Formular.F??	Rez. entwickeln	Warteschlange	Batchfile drucken	Laufwerk ändern	T.Lesen
Rezeptur.REZ	Ans. entwickeln	Naßgranulierung	Formular erstellen	Ordner anlegen	T.Rez.+Ans.
Ansaetze.REC	Rez. berechnen	WS-Granulierung	Druckeranpassung	Ordner löschen	T.Abl.planen
Berichte.REP	Ans. berechnen	Tablettierung	-Inline - Editor :	Datei löschen	Programmende
Prozeßordner	Dat. verketten	Filmcoating	The Sparrow	Disk-Verzeichnis	
	Ablauf planen	Dragierung	»EDITOR« laden	freier Speicher	
		Pelletierung		Schreibschutz	
		Retardierung		Datum + Uhrzeit	
		Freiformular		Programmende	
		Vermahlung			
		Kapselabf.			
		Suspensionsans.			
		Lösungsans.			
		Sterilisation			
		Ampullenabf.			
		Infusionsabf.			
		Physik.Meßwerte			
		Hygienekontrolle			
		Stabilitätsdaten			

Bild 1: Die Pull-Down-Menüs von Galenius

Es ist auch möglich, Rezepturen aus Ansätzen und umgekehrt berechnen zu lassen. Ebenso können bereits vorhandene Daten miteinander verkettet werden. Es folgt der wichtigste Programmpunkt, die Ablaufplanung. Hier wird aus dem Formularsatz, der Ansatzdatei und den Prozeßmodulen der Report zusammengestellt. Nach Selektion dieses Programmpunktes erscheint eine Maske (siehe Bild 3). Hier kann man die Prozesse, die für die Herstellung eines Medikamentes benötigt werden, selektieren. Es können jeweils zehn Prozesse ausgewählt werden. Eine weitere Möglichkeit ist dabei die Warteschlange (Bild 4). Auch hier können bis zu zehn Prozesse zusammengestellt werden.

Das Zusammenbinden der einzelnen Files dauert einige Zeit. In einer Box läßt sich die Tätigkeit des Computers mitverfolgen. Dort steht dann die Länge des kompletten Files im ersten Feld, im zweiten die bereits gelesenen Bytes. Es folgen dann noch Zeilenzähler für die Batchfiles der Ansatzdatei und der Prozeßmodule. Während dieses Ablaufes blendet das Programm mittels Fileselector-Box die einzelnen Prozeßordner ein. Hier ist es möglich, die speziellen Module für das Medikament anzuwählen. Hat sich Galenius zurückgemeldet und ist kein Fehler aufgetreten, kann der Bericht gespeichert werden. Dabei ist positiv zu bemerken, daß Galenius während des Speicherns die Formulare auf dem Bildschirm in Mikroschrift anzeigt, so daß man eine erste Kontrolle über deren Richtigkeit hat. Sollte eine Korrektur notwendig sein, ist dies natürlich ebenfalls möglich.

|F1 |SF2leer|F3 |F4 |F5 |F6 |F7 |F8 |SF9 Scrol⇧|F10 |
|Titel|F2Zeile|Korr. |Löschen|Mark.LM|Abschl.|Fußnote|Speich| F9 Scrol⇧|Ausgang|

Titelzeile B:\REZEPTUR\SCHWIEG.REZ

Bestandteil	Menge in mg/Dosis
1. Kaliumcyanid	1296.762
2. Arsenicum	3241.904
3. Mercurium	5609.791
4. Phosgenium (COCl2)	2286.191
5. Omnitablettol	30704.722
6. E 605	159.502
7. Aqua Destilata	334.565
8. Dispergin	879.204
9. Methanol absolut	13723.629
10. Apis Melefica	754.715
11. Patentcoating EF	20474.57
12. Butolinus Extrakt	59.248
13. Lactose	20475.199
Summe der Feststoffe	100000

1296

Fußnote Rezeptur SchwiegermutterEX-Forte

Bild 2: Die Rezeptureingabemaske mit einem "todsicheren" Medikament

Formular 01 bis 10 -> F-Taste gleicher Nummer oder Klick F-Feld
Formular 11 bis 20 -> Shift + F-Taste oder Klick F-Feld

Herstellungsvorgang planen
Batchdatei zusammenstellen

Verfügbare Formulare	Verfügbare Formulare	Zusammenstellung
Titelseite	Vermahlung	TITELSEI.F01
Einwaage	Kapselfüllung	EINWAAGE.F02
Naßgranulierung	Suspensionsans.	VERMAHLE.F11
WS-Granulierung	Lösungsans.	WSGRANUL.F04
Tablettierung	Sterilisation	TABLETTE.F05
Filmcoating	Ampullenabf.	MESSWERT.F18
Dragierung	Infusionsabf.	
Pelletierung	Physik.Meßwerte	
Retardcoating	Hygienekontrolle	
Freiformular	Stabilitätsdaten	
Bericht-Batch binden Maus l. oder >> b <<	*.REC-Datei aufrufen Maus l. oder >> d <<	Korrektur UNDO oder Maus l.
REC in Bericht fügen Maus l. oder >> + <<	Bericht speichern Maus l. oder >> s <<	Ausgang Maus l. oder >>ESC<<

Bild 3: Ablauf planen

Galenius lädt die Dateien nacheinander ein, und diese können auf die benötigten Bedürfnisse geändert werden. Wählt man Ausgang an, wird der nächste Prozeß geladen.

Was nun folgt, ist fast nur reine Mausarbeit. Man klickt *Bericht-Batch binden* an. Dies bewirkt, daß Galenius die gewählten Formulare aus dem Ordner holt und den noch leeren Formularsatz abspeichert. Es folgt *x.REC-Datei aufrufen*. Hier wird ein Ansatz geladen, damit dieser den Formularen beigefügt werden kann. Er wird dann mit *REC in Bericht einfügen* in den Bericht eingefügt.

Jetzt ist das Batchfile fertig und kann ausgedruckt werden. Dann wird unter dem Menüpunkt *Berichte, Batchfile drucken* angewählt. Bevor man jedoch den Report ausdruckt, ist der Drucker zu konfigurieren. Galenius bietet hierfür ein Anpassungsprogramm für den NEC P6 an, das vom Programm aus aufgerufen werden kann. Leider befinden sich keine anderen Druckertreiber auf der Diskette, aber ein Versuch mit einem Star NL10 ergab keine Probleme und der Ausdruck erfolgte in ordentlicher Form. Sollte ein ansprechender Ausdruck nicht möglich sein, steht es dem Anwender frei, andere Treiber, wie sie z.B. mit Public Domain zu bekommen sind, zu verwenden. Diese müssen nur in *DRUCKER.PRG* umbenannt sein. Wichtig ist weiterhin, daß der Drucker auf den deutschen Zeichensatz eingestellt ist, da sonst die Umlaute nicht mitgedruckt werden.

Schwarz auf Weiß sollte es sein.

Nach Anwahl des Menüpunkts *Drucken* fragt Galenius nach, welche Datei ausgegeben werden soll. Während des Ladens zeigt das Programm an, wieviele Zeilen im Speicher sind. Es erscheint dann der Report auf dem Bildschirm und man kann ihn noch einmal anschauen. Wird die rechte Maustaste gedrückt, erscheint eine kleine Box, die abfragt, ob es sich um die korrekte Datei handelt. Danach folgt noch eine letzte Abfrage ob gedruckt werden soll, und wird diese mit *Ja* beantwortet, wirft

der Drucker regelrecht mit Papier um sich. Der Druckvorgang läßt sich zum Ende jeder Seite abbrechen. In Bild 5 ist ein Teil eines mehrseitigen Reportes zu sehen.

Selbst ist der Mann/die Frau...

...denn meist werden Formulare gebraucht, die nicht vorgefertigt sind. Deshalb können Sie diese auch selbst herstellen . Hier gibt es zwei Wege, die zum Ziel führen. Die erste Möglichkeit ist, vorhandene Formulare zu verändern. Das bietet den Vorteil, daß man nicht mit den Sonderzeichen zu kämpfen hat, mit denen Galenius zu Steuerzwecken arbeitet. Hier kommt auch der eingebaute Editor *The Sparrow* zum Tragen. Vom Funktionsumfang erfüllt er zwar nur die Grundbedürfnisse, hat aber den Vorteil, daß er alle Steuer- und Sonderzeichen anzeigt. Weiterhin verfügt er über eine Font-Box, mit der diese Zeichen auch genutzt werden können. Der andere Fall liegt vor, wenn gänzlich neue Formulare erstellt werden müssen. Hier muß eng nach den Vorgaben aus dem Handbuch gearbeitet werden. Die Zeilenlänge sollte 74 Zeichen betragen und die Länge des Formulars mindestens 38 Zeilen lang sein. Den Abschluß eines Formulars bildet dann das Zeichen CHR$(218) und CHR$(219) zeigt das Dateiende an. Bild 6 zeigt, wie man sich ein Formular vorzustellen hat. Die Funktion der Sonderzeichen wird in einer Tabelle genau erklärt. Wer einen anderen Editor besitzt, der auch Sonderzeichen verarbeitet wie z.B. Tempus, kann auch diesen verwenden.

Diskettenoperationen

Unter diesem Menüpunkt sind Änderungen der Laufwerke, Anlegen und Löschen von Ordnern, Anzeigen des Inhaltsverzeichnisses, Anzeige des freien Speichers sowie von Datum und Uhrzeit möglich. Interessant ist die Umschaltung zwischen Nur-Lese- und Schreib/Lese Status von Dateien, womit man Dateien vom Programm aus vor ungewollten Überschreibungen und Löschungen schützen kann. Als Nothilfe stehen einige Menüs mit Kurzbeschreibungen zur Verfügung.

Fazit

Bis auf das oben erwähnte kleine Problem mit TOS 1.2, das ja behoben wurde, hinterließ GALENIUS einen sehr guten Eindruck. Das Programm ist ausgereift und bietet eine Vielzahl von Funktionen, die bei der Erstellung von Produktionsabläufen die Arbeit sehr erleichtern. Einzig negativ mag sein, daß GALENIUS alle ACC-Einträge deselektiert und diese dadurch nicht aufrufbar sind. Durch den Einsatz des Computers ist eine sehr große Flexibilität gegeben, wie sie in der Forschung und bei öfters durchzuführenden Produktionsänderungen benötigt wird. Die Einsatzbereiche von GALENIUS liegen deshalb vor allem in der Forschung und Entwicklung sowie der Herstellung in der Pharmazeutischen Industrie. Weitere Einsatzbereiche mögen für Apotheker gegeben sein, die Medikamente selbst herstellen und für Studenten der Pharmazie, als flexible Hilfe für durchzuführende Praktika.

Rainer Esser

Preis: DM 640.-

Bezugsquelle:

Softchemie
Postfach 1048940

Bild 4: Wahlweise mit Warteschlange

Bild 5: Ein Formulargerüst

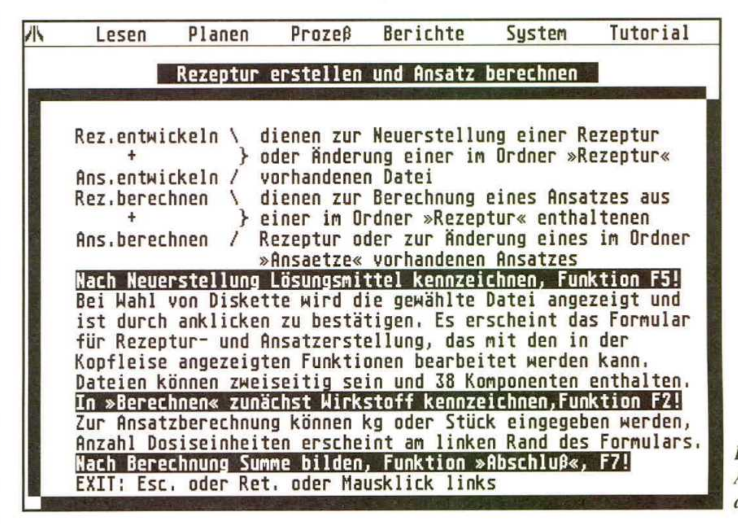


Bild 6: Als letzte Hilfe, das Hilfsmenü

Einkaufsführer

Hier finden Sie Ihren Atari Fachhändler

1000 Berlin



Kurfürstendamm 121a, 1000 Berlin 31 (Halensee)
Telefon 030/891 1082



neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Breite Straße 50 • ☎ 030-3 33 20 08

DATAPLAY

Bundesallee 25 · 1000 Berlin 31
Telefon: 030/861 91 61

Computare

Keithstr. 18-20 • 1000 Berlin 30
030/21 390 21
186 346 com d



Vertragshändler

UNION ZEISS

Kurfürstendamm 57 • 1000 Berlin 15
Telefon 32 30 61

1000 Berlin

Ihr Atari-Händler in Berlin

COMPUTERSHOP
Radtke u. Kögel

Riesen Software Angebot

Fürbringerstr. 26 · 1000 Berlin 61
Tel. (030) 6 91 46 29 · BTX (030) 6 91 76 66

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Tempelhofer Damm 195 • ☎ 030-7 52 30 55

HD *Der ATARI ST Hardware Spezialist*
Computertechnik

PUBLIC DOMAIN SERVICE

Sämtliche verfügbaren ST COMPUTER - PD's. vorrätig und weitere Serien für den ST, z.B. PD-Pool und ST-Vision.

1000 Berlin 65 * Pankstr. 42
Tel.: 030/465 70 28

SERVICE STATIONEN
Auch hier alle PD s vorrätig!
1/44, Lahnstrasse 94
1/20, Schönwalder Str. 65



neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft



Johannisth. Chaussee 301 • ☎ 030-6034056

1000 Berlin

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Buckower Chaussee 100 • ☎ 030-7216004

COMPUTER-STUDIO

Schlichting

... die etwas andere Computerei

Ihr Spezialist in Berlin für Hardware + Zubehör Eigenes Softwarestudio über 1000 verschiedene Titel am Lager

ATARI-Fachmarkt
NEC-Fachhandel · MS-DOS Fachmarkt

Katzbachstraße 6 + 8 · 1000 Berlin 61
☎ 030/7864340

2000 Hamburg

Bit Computer Shop

Osterstraße 173 · 2000 Hamburg 20
Telefon: 040/494400

Createam
Computer Hard & Software

Bramfelder Chaussee 300 · 2000 Hamburg 71
Telefon Sa. Nr. 040/641 50 91

RADIX Bürotechnik

Heinrich-Barth-Straße 13
2000 Hamburg 13
Telefon (0 40) 44 16 95

NEU: Software Shop



Planen • Beraten • Realisieren



ATARI Fachhändler

Münsterstraße 9, D-2000 Hamburg 54
Telefon 040/56 19 09-0
Telefax 040/56 19 09-80

2000 Norderstedt



2120 Lüneburg

Sienknecht
Bürokommunikation
Beratung - Verkauf - Werkstatt

Heiligengeiststr. 20, 2120 Lüneburg
Tel. 04131/46122, Btx 402422
Mo.-Fr. 9°°-18°° und Sa. 9°°-13°°

2210 Itzehoe

Der Computerladen
Inhaber Ulrich Bubel · Martin Kopplow

Coriansberg 2 · 2210 Itzehoe
Telefon (04821) 3390/91

2300 Kiel



Die Welt der Computer
Dreiecksplatz Nr. 7
2300 Kiel 1 · ☎ 0431/567042

2800 Bremen

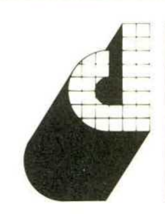


Faulenstraße 48—52
2800 Bremen 1
Telefon (0421) 170577

2940 Wilhelmshaven

Radio Tiemann
ATARI-Systemfachhändler
Markstr. 52
2940 Wilhelmshaven
Telefon 04421-26145

3000 Hannover



DATALOGIC COMPUTERSYSTEME
ATARI ST-COMPUTER HARDWARE SOFTWARE
BERATUNG SERVICE VERKAUF
CALENBERGER STR. 26
3000 HANNOVER 1
TEL.: 0511 - 32 64 89

3000 Hannover



Am Schiffgraben 19 · 3000 Hannover 1
Telefon 0511-326736

3400 Göttingen

Büroeinrichtungs-Zentrum
Wiederholdt
3400 Göttingen-Weende
Wagenstieg 14 – Tel. 0551/3857-0

3500 Kassel

Hermann Fischer GmbH
autorisierter ATARI-Fachhändler

Rudolf-Schwander-Str. 5-13
3500 Kassel
Telefon (0561) 700000

4000 Düsseldorf

BERNSHAUS GmbH
Bürotechnik – Bürobedarf
Cäcilienstraße 2
4000 Düsseldorf 13 (Benrath)
Telefon 0211-719181

Hard und Software
Werner Wohlfahrtstätter

Atari	**Ladenlokal**
Public Domain	Irenenstraße 76c
Atari Spiele	4000 Düsseldorf-Unterrath
Atari Anwender	Telefon (0211) 429876

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Erkrather Str. 169-179 • ☎ 0211-736214

HOCO
EDV ANLAGEN GMBH
Ellerstraße 155
4000 Düsseldorf 1
Telefon 0211/785213

4010 Hilden

Beachten Sie unsere Anzeige in diesem Heft!



Computer · Drucker · Software · Bücher · Service
Gustav-Mahler-Straße 42-44
Tel. (02103) 31880+41226

4150 Krefeld



DTP-CENTER
MATRIX *MatScreen*
ROLAND *Schnittplotter*
DATACOPY *Scanner*
Digitizer

C.O.P Computer-Service GmbH
Tannenstr. 103
4150 Krefeld
☎ (02151) 77 30 41
FAX (02151) 77 05 86

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Ostwall 138 • ☎ 02151-1503

4200 Oberhausen

Redakteur • Spectre 128



für TOS, DOS, UNIX, MAC, AMIGA und andere Systeme
by COMPUTER MAI

dBMAN-Vertretung für NRW und BENELUX
ISYS-COMPUTER GbR
Tel.: 0208/655031 · Telefax: 0208/650981
Max-Eyth-Straße 47 · 4200 Oberhausen 11

4300 Essen

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Limbecker Platz 4300 Essen 1
Tel.: (0201) 176399

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Limbecker Str. 12-16 • ☎ 0201-233515



4300 Essen
Limbecker Straße 12-16
Tel. (0201) 233515 · Fax 233834

4320 Hattingen



4422 Ahaus

ATARI · Epson · Fujitsu Molecular · NCR · Tandon · Schneider · Star

OCB

OCB-Computershop
Wallstraße 3
4422 Ahaus
Tel. 0 25 61/50 21

OCB-Hard- und Software
Wessumerstraße 49
4422 Ahaus
Tel. 0 25 61/50 21

4430 Steinfurt

CBS GmbH
COMPUTERSYSTEME
Tecklenburger Str. 27
4430 Steinfurt-Burgsteinfurt
☎ 02551/2555

4500 Osnabrück

Heinicke-Electronic
Meller Str. 43 · 4500 Osnabrück
Fax (0541) 58 66 14
Telefon (0541) 58 66 46
Wir liefern Micro-Computer seit 1978

4520 Melle

CBS GmbH
COMPUTERSYSTEME
4430 Steinfurt Tel. 02551/2555
Haferstraße 25 4520 Melle
Tel.: 05422/44788

4600 Dortmund

Elektronik
Computer
Fachliteratur
ATARI-System-Fachhändler
4600 Dortmund 1, Güntherstraße 75, Tel. (02 31) 57 22 84



ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Kampstraße 1 · 4600 Dortmund
Telefon (02 31) 5 43 91

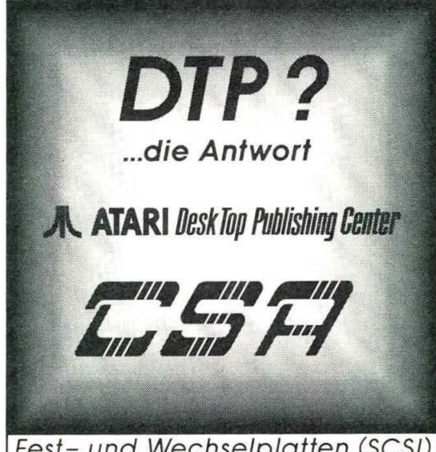
4600 Dortmund

cc Computer Studio GmbH
Atari-Systemfachhändler
PCs von Tandy Schneider Peacock
Drucker von Star Brother NEC
Elisabethstr. 5
4600 Dortmund 1
Tel. 0231/528184 Tx 822631 cccsd Fax 0231/528131

4650 Gelsenkirchen-Horst

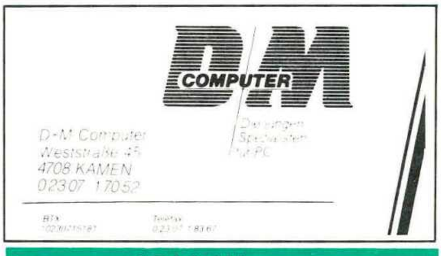
MENTIS GmbH
Hard- und Software, Literatur
Bauteile, Service, Versand
Groß- und Einzelhandel
Poststraße 15 · 4650 Gelsenkirchen-Horst
Telefon (02 09) 5 25 72

4650 Gelsenkirchen



Fest- und Wechselplatten (SCSI)
Großmonitore, Scanner, Software
ComputerSysteme und Anwendungen
Hüttenstraße 56 – 4650 Gelsenkirchen
Tel. (0209) 203420 FAX: (0209) 271584

4708 Kamen



4712 Werne

Vogler & Trümper
Hard- und Software



Lünener Straße 14
4712 Werne
Tel. (0 23 89) 5 14 95

4800 Bielefeld

hardware software organisation service
CSF
CSF COMPUTER & SOFTWARE GMBH
Heeper Straße 106 – 108
4800 Bielefeld 1
Tel. (05 21) 6 16 63

5000 Köln



5090 Leverkusen

Rolf Rocke
Computer-Fachgeschäft
Auestraße 1
5090 Leverkusen 3
Telefon 0 21 71 / 26 24

5210 Troisdorf

LOGITEAM
Computerhandelsgesellschaft mbH
Kölner Straße 132
5210 Troisdorf
Tel. (0 22 41) 7 18 97
FAX (0 22 41) 7 58 58



5253 Lindlar



5300 Bonn



5414 Vallendar



ATARI-Systemfachhändler für Mayen-Koblenz
Ihr autorisierter Fachhändler für GTC- Personalcomputer, Star, Epson und NEC
Wir schreiben BERATUNG und SERVICE groß!
Zentrale: 5414 Vallendar, Rheinstr. 117, TEL. 0261/61727
5419 Dierdorf, Hauptstraße 50
5500 Trier, Ehrangerstr. 31

5430 Montabaur

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Industriestr. 20-24 • ☎ 0 26 02-34 67

5500 Trier



Güterstraße 82 · 5500 Trier

☎ 06 51 / 20 97 10

Fordern Sie unsere Zubehör-Liste an!

5600 Wuppertal

COMPUTER FINKE



ATARI - SYSTEMFACHHÄNDLER

KIPDORF 22 · 5600 WUPPERTAL 1 · TEL 0202 45 32 33

HARDWARE · SOFTWARE · ZUBEHÖR · SERVICE · SCHULUNGEN



... wir machen Spitzentechnologie preiswert.

MEGABYTE

Computer Vertriebs GmbH

Friedrich-Engels-Allee 162
5600 Wuppertal 2 (Barmen)
Telefon (02 02) 8 19 17

5630 Remscheid

COM SOFT

Nordstraße 57 · 5630 Remscheid
Telefon (0 21 91) 2 10 33

5650 Solingen

MegaTeam

Computer-Vertriebs-OHG
Kölbach – Finke

Hardware – Software – Zubehör – Service

Rathausstraße 1-3 · 5650 Solingen 1
Telefon (02 12) 4 58 88 · Fax (02 12) 4 73 99

5800 Hagen



Vertragshändler Axel Böckem

Computer + Textsysteme

Eilper Str. 60 (Eilpezentrum) · 5800 Hagen
Telefon (0 23 31) 7 34 90

5860 Iserlohn

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Theodor-Heuss-Ring 27 • ☎ 0 23 71-2 36 43

5900 Siegen



Siegen · Weidenauer Str. 72 · ☎ 02 71/7 34 95

6000 Frankfurt

WAIZENEGGER

Büroeinrichtungen

Kaiserstraße 41
6000 Frankfurt/Main
Tel. (0 69) 2 73 06 - 0

Bei uns werben bringt

GEWINN

Sprechen Sie mit uns.
Heim Verlag ☎ 0 61 51 / 5 60 57

BUF

Eickmann Computer

Der **Atari-System-Fachhändler !**

z.B.: Festplatten von 30 MB bis 110 MB für Atari ST und Mega ST, Zusatzaufrüstungen für Ihre Festplatten bis 110 MB, Umrüstung Ihres SM 124 in einen EM 124 Multisync für alle Auflösungen, Slotkit für PC 1...

besuchen Sie unser Fachgeschäft:
In der Römerstadt 249
6000 Frankfurt 90-Praunheim
Telefon (069) 763409

6100 Darmstadt

Heim

Büro- und Computermarkt

Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon (0 61 51) 5 60 57

6108 Weiterstadt

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft



Friedrich-Schäfer-Str.2 • ☎ 06151-89 58 34

6123 Bad König



! Vor(Ver)führungen !

CALAMUS	OUTLINE ART	ARABESQUE
Repro Studio ST	PEGASUS +	DRAFTER CAD
TmS Graphics	TmS CRANACH	TmS Vektor
Sherlook OCR	SIGNUM ! 2	SCRIPT
ADIMENS PLUS	BSS Plus	Reprok
Eickmann Turbo 16 MHz		PC SPEED 1.4
PANASONIC Scanner 506 U	LACOM Wechselplatte	

Speichererweiterung 2 MB: 598.– / 4 MB: 998.–
Laufwerk 5 1/4" 40/80 Track: 298.–
Hard und Software aller Hersteller

6200 Wiesbaden



wir machen Spitzentechnologie preiswert

COMPUTERCENTER

DER SYNTHESIZER • STUDIO JACOB GMBH

Computer, Software, Zubehör
MIDI - Equipment, eigener Service

6200 WIESBADEN, MAINZER STR. 137
Tel. (06121) 71 94 90 BBS: (06121) 70 17 39

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Didierstraße 27a • ☎ 0 61 21 - 60 30 21

6236 Eschborn

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft



Philipp-Helfmann-Str.2-4 • ☎ 0 61 96-4 28 21

6240 Königstein

KFC
COMPUTERSYSTEME

Wiesenstraße 18
6240 Königstein
Tel. 0 61 74 - 30 33
Mail-Box 0 61 74 - 53 55

6250 Limburg



6300 Gießen



6400 Fulda

Schneider ATARI Commodore

WEINRICH

BÜRO · ORGANISATION

Ronsbachstraße 32 · 6400 Fulda

Telefon (06 61) 4 92 - 0

6457 Maintal

LANDOLT - COMPUTER



Beratung - Service
Verkauf - Leasing
Finanzierung



6457 Maintal-Dörnigheim Robert-Bosch-Straße 14
Tel. (06181) 4 52 93 Fax (06181) 43 10 43
Mailbox (06181) 4 88 84 Btx *2 98 99#

6500 Mainz - Hechtsheim

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Alte Mainzer Str. 164 • ☎ 06131-83 45 17

6520 Worms



Computersysteme
GmbH

6520 Worms · Friedrichstraße 22

Telefon 0 62 41 / 67 57 - 58

6581 Niederbrombach

INFODAT GmbH

ATARI Computersysteme
Protar Center

Beratung - Softwareentwicklung - Individuallösungen

Unter Hochkastell 3
6581 Niederbrombach
Telefon: (06787) 1425

6630 Saarlouis

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Ahornweg 1 - 3 • ☎ 0 68 31 - 8 00 95

6700 Ludwigshafen

MKV Computermarkt

Bismarck-Zentrum
6700 Ludwigshafen
Telefon 06 21 - 52 55 96

6720 Speyer

THEILLE
Computersysteme
Gilgenstraße 4 · 6720 Speyer
Telefon (0 62 32) 7 72 16

Etzkorn

DIE FACHLEUTE FÜR COMPUTER

AUTORISIERTER ATARI - VERTRAGSHÄNDLER
HARDWARE, SOFTWARE & ZUBEHÖR FÜR
• Apple • Acorn • Jonathan • MS/Dos •

6720 Speyer Auestraße 20
Telefon 06232 / 32428 oder 32435
Mo.-Fr. 9 - 12 u. 15 - 18.30, Sa. 9 - 14 Uhr

6750 Kaiserslautern

INFODAT GmbH

ATARI Computersysteme
Protar Center

Beratung - Softwareentwicklung - Individuallösungen

Schubertstr. 16 6750 Kaiserslautern
Tel: (0631) 63597 Fax: (0631) 63589

6800 Mannheim



Computersysteme + Textsysteme

6800 Mannheim 24

Casterfeldstraße 74-76

☎ (06 21) 85 00 40 · Teletex 6 211 912

6900 Heidelberg

JACOM FAMILA-CENTER

Hardware · Software
Schulung · Service

Hertzstraße 1 · 6900 Heidelberg 1
Telefon (0 62 21) 30 24 37

7000 Stuttgart



7030 Böblingen

Verkauf - Service - Software



Norbert Hlawinka
Sindelfinger Allee 1
7030 Böblingen
Tel. 0 70 31 / 22 60 15

COMPUTER CENTER

7047 Jettingen

Verkauf - Service - Software

Norbert Hlawinka
Heilbergstraße 3
Im Multi-Center
7047 Jettingen
Telefon (0 74 52) 7 76 15



COMPUTER SHOP

7100 Heilbronn

Computer-Welt



Am Wollhaus 6
7100 Heilbronn
Tel. 0 71 31 - 6 84 01 - 02

Hohmann & Co

Mönchseestraße 99
7100 Heilbronn
Telefon: 07131/60048

7150 Backnang

Computer-Fans finden bei uns alles von:



Bei uns werben bringt
GEWINN
Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057

BUF

7312 Kirchheim/Teck

Comp & Phone

Computerfachhandel

Alleenstraße 66
7312 Kirchheim/Teck
Tel.: 07021/3949 - Fax: 07021/53933

7410 Reutlingen

MKV GMBH

Listplatz 2
7410 Reutlingen
Telefon 0 71 21 - 3 66 47

7475 Meßstetten

Ihr ATARI-Systemhändler im Zollern-Alb-Kreis
HEIM + PC-COMPUTERMARKT
HARDWARE · SOFTWARE · LITERATUR
SCHEURER
ATARI COMMODORE CUMANA DATA-BECKER MULTITECH RITEMAN SCHNEIDER THOMSON
7475 Meßstetten 1 · Hauptstraße 10 · 07431/61280

7500 Karlsruhe

MKV GMBH

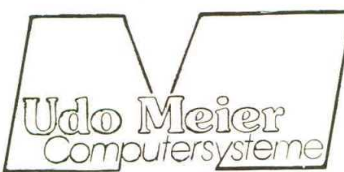
Kriegsstraße 77
7500 Karlsruhe
Telefon (0721) 84613

ERHARDT Am Ludwigsplatz
Am Ludwigsplatz · 7500 Karlsruhe 1 · Tel. (0721) 1608-0

7600 Offenburg

FRANK LEONHARDT ELECTRONIC
Ihr Fachgeschäft für Microcomputer · Hifi · Funk
In der Jeuch 3
7600 Offenburg
Telefon 0781/57974

7700 Singen



Ringstraße 4
Telefon (07731) 68222

7730 VS-Schwenningen

BUS BRAUCH & SAUTER COMPUTER TECHNIK
Villinger Straße 85
7730 VS-Schwenningen
Telefon 07720/38071-72

7750 Konstanz

ATARI ★ PC's ★ SCHNEIDER
computer - fachgeschäft
rösler
Rheingutstr. 1 · ☎ 07531-21832

7890 Waldshut-Tiengen

hettler-data
service gmbh
Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon 07751/3094

7910 Neu - Ulm

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Wegenerstraße 1 • ☎ 0731-81018

7918 Illertissen

bidtech gmbh
technische Informationsysteme Computerladen
Marktplatz 13
7918 Illertissen
07303/5045

7930 Ehingen

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Thalstraße 12 • ☎ 07391-8028

8000 München

Ihr Spezialist für:
dBMAN - Komplettlösungen
vortex Massenspeicher
Vortex, dBMAN sind eingetragene Warenzeichen zugunsten Dritter
ABAC München
Kellerstraße 11, 8000 München 80
Tel. 089/ 448 99 88

ludwig
COMPUTER + BÜROTECHNIK
COMPUTER · SOFTWARE · PERIPHERIE
BERATUNG · TECHN. KUNDENDIENST
INGOLSTÄDTER STRASSE 62L
EURO-INDUSTRIE-PARK · 8000 MÜNCHEN 45
TELEFON 089/3113066 · TELETEX 898341

City Studio
Rindermarkt 6, 8000 München 2
Tel. 089/318195-0

schulz computer
Schillerstraße 22
8000 München 2
Telefon (089) 597339
Beratung · Verkauf · Kundendienst

8000 München

T.S. Service
Szemere Hard & Software
Schleißheimerstr. 220
D-8000 München 40

Tel. 089/3089408 Fax 089/3085636 ATARI
Festplatten / Rechner / Zubehör / Service
Bürozeiten: Mo. bis Fr. 9.00-12.00 13.00-18.30 Samstag 9.00-14.00

8032 Gräfelfing



Pasinger Straße 94 8032 Gräfelfing
Telefon 089-8548823
Fax 089-8541764

8150 Holzkirchen


MÜNZENLOHER GmbH
Tölzer Straße 5 * 8150 Holzkirchen
Tel:(08024)1814 * Fax:(08024)4879
ATARI-SCHNEIDER-NEC
PANASONIC-PHILIPS
TOSHIBA PORTABLE-LUCKY GOLDSTAR
Hard- und Software in großer Auswahl
Service und Beratung sind bei uns Inklusive

8200 Rosenheim



COMPUTER + BÜROTECHNIK
Kufsteiner Str. 11, 8200 Rosenheim
Tel. (08031) 38 00 30
Fax (08031) 1 53 34

Autorisierter ATARI System-Fachhändler
Fischer & Bach
Computer GmbH
Münchner Straße 41 · 8200 Rosenheim
Tel.: 0 80 31/1 47 55 · Fax: 0 80 31/1 76 67
Hardware · Software
Schulung · Schnellreparatur

8390 Passau

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Kohlbruck 2a • ☎ 0851-52007

8400 Regensburg

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft


Dr.Gessler-Str. 8 • ☎ 0941-95085

8400 Regensburg



im ELEKTROLAND REGENSBURG
Dr.-Gessler-Straße 8
Tel. (09 41) 9 50 85 · Fax 94 79 61

8423 Abensberg

WITTICH
COMPUTER GMBH
Tulpenstr. 16 · 8423 Abensberg
☎ 0 94 43 / 4 53



8480 Weiden

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Obere Bauscherstr.21 • ☎ 09 61-4 61 05

8500 Nürnberg

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Färberstraße 20 • ☎ 09 11 - 20 80 84

8520 Erlangen

Computerservice
Decker
Meisenweg 29 - 8520 Erlangen
Telefon 0 91 31 / 4 20 76

Zimmermann
elektroland

8520 Erlangen
Nurnberger Straße 88
Tel. (09131) 34568

8500 Nürnberg
Hauptmarkt 17
Tel. (0911) 20798



8520 Erlangen

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Nürnberger Str. 88 • ☎ 0 91 31-3 28 56

8600 Bamberg



neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Ludwigstr. 2 • ☎ 09 51-2 51 37-38

8700 Würzburg

SCHOLL
BÜROTEAM
Hardware · Software
Service · Schulung
computer center
am Dominikanerplatz
Ruf (09 31) 3 08 08 - 0

8720 Schweinfurt

Uhlenhuth GmbH
Computer + Unterhaltungselektronik
Albrecht-Dürer-Platz 2
8720 Schweinfurt
Telefon 0 97 21 / 65 21 54

8900 Augsburg

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Eichleitnerstr. 14 • ☎ 08 2-5 70 57 50

Adolf & Schmoll
Computer
Unser Plus: Beratung u. Service
Schwalbenstr. 1 · 8900 Augsburg-Pfersee
Telefon (08 21) 52 85 33 oder 52 80 87
Computer Vertriebs- und Software GmbH

8906 Gersthofen

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Dieselstraße 5a • ☎ 08 21 - 49 50 29

8940 Memmingen

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Schlachthofstr. 53 • ☎ 0 83 31-4 70 97

8960 Kempten

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Bleicherstraße 30 • ☎ 08 31 - 7 10 86

ÖSTERREICH

A-1030 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m.b.H.
A-1030 Wien
Landstraßer Hauptstraße 2
Hilton-Einkaufspassage

A-1040 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m.b.H.
A-1040 Wien · Paniglgasse 18-20
Tel. (02 22) 5 05 78 08, 5 05 88 93

A-1180 Wien

Ihr kompetentes Atari-Fachgeschäft in Wien



Tel. (0222) 48 52 56
A-1180 Wien - Schulgasse 63

SCHWEIZ



Ihr Computer Spezialist

5000 Aarau, Bahnhofstrasse 86, Tel. 064/22 78 40
4102 Basel-Binningen, Kronenplatz, Tel. 061/47 88 64
5430 Wettingen, Zentralstrasse 93, Tel. 056/27 16 60
8400 Winterthur, St. Gallerstrasse 41, Tel. 052/27 96 96
8021 Zürich, Langstrasse 31, Tel. 01/241 73 73

Grösste Auswahl an Peripherie, Software, Literatur und Zubehör.

CH-1205 Geneve

PIMENT ROUGE
INFORMATIQUE S.A.
8, RUE DES MARAICHERS
1205 GENEVE TEL. 022/28 56 24

Bei uns werben bringt

GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057

BUF

CH-1700 Fribourg

FRIDAT SA INFORMATIQUE
ehem. Softy Hard's Computershop

VOTRE SPECIALISTE
Rte des Grives 4
1700 Granges-Paccot/Fribourg
Tel. 0041 (0)37 26 66 28
Fax. 0041 (0)37 26 61 06

CH-2503 Biel



URWA ELECTRONIC
Computer Hard- und Software

Ihr ATARI ST Spezialist
in der Schweiz.
☎ 0 32 / 41 35 35

Bözingenstraße 133, 2504 Biel

CH-3006 Bern

C.A.D. Atelier Dellsperger
Brunnadernstrasse 18, CH-3006 Bern

Grosse Auswahl an:
Hardware
Software
Zubehör
Support
Schulung
Leasing
Service
Occ. /Scann- & Plottservice.

ATARI
MATRIX MatScreen
Roland Plotter
EPSON
Technologie, die Zeichen setzt

Berns Nr. 1 für freundliche & kompetente Beratung und Support.

C.A.D. Atelier Dellsperger
Brunnadernstrasse 18, CH-3006 Bern
Tel: 031 43 00 35

CH-4313 Möhlin

BCR Computerdienst
Bahnhofstrasse 63
CH-4313 Möhlin

Computersysteme
EDV-Beratung
Installationen
CAD Anlagen
Datenpflege + Service

ATARI
NEC
star

Tel. 061 88 30 32 FAX 061 88 30 03

CH-4500 Solothurn

Fluri Informatik



Hard- & Software, Zubehör
ATARI Schulungszentrum
Desktop Publishing
Systemlösungen

Schänzlistr. 4 4500 SOLOTHURN 1
Tel. 065 / 23 68 58 Fax. 065 / 23 16 57

CH- 4625 Oberbuchsiten

STECTRONIC M. Steck
Electronic-Computer-Shop

Hauptstr. 104/137
CH-4625 OBERBUCHSITEN
Tel. 0 62 / 63 17 27 + 63 10 27

CH-5400 Baden

Eine neue Dimension im ATARI Hard - und Softwarebereich:
ABAKU Computer + Communication AG
Mäderstrasse 1
5400 Baden / Schweiz
Tel. 056 - 22 10 31 • Fax. 056 - 22 10 32

LUXEMBURG



Bei uns werben bringt

GEWINN



Sprechen Sie mit uns.

Heim Verlag
☎ 0 61 51 / 5 60 57

Kleinanzeigen

Suche Hardware

Suche defekten Drucker, mindestens 200 cps Draft
Tel.: 0 74 83 / 394 ab 14 Uhr

Suche autom. Einzelblatteinzug (CSF) für NEC P7 (alt).
Rainer Seidel, Tel.: 0 81 53 / 23 67

Kleinanzeigen

Biete Software

Verkaufe GCR-Spectre mit Org. Mac-Roms DM 950,-
Liest Disketten im Mac-Format
V. Brielmayer, 7014 Kornwestheim
Langestr. 53 Tel. 0 71 54 / 2 73 89

Lattice C V. 3.03 + Metacomco Pascal V. 1.35 zus. 150,-
Tammen / Landeck 05 61 / 10 47 95

2000 Artikel aus ST-COMP. ab 9/86 (ohne RELAX) c*t ab 1/87, mc 1/87-5/89 & versch. aus 68000, ST-MAGAZIN als **ADIMENS & ASCII-Datei**
DM 30,- Vorkasse von **M. Emmer, Möhlestr. 10, 8050 Freising**

HP-9000 3.5"-Diskette (LIF) am ATARI ST lesen + beschreiben (HP-86 lesen). **HP-150<>ST** auf Anfrage. Info: Dr. M. Poech
07 11 / 29 71 49, 18-21 h

Für Lehrer u. Schüler u. andere
Datenmanager für alle Bereiche 89,- DM, weiter Vokabeltrainer Engl./Deutsch - Deutsch/Engl. mit Editier- u. Eingabemodus, 5 kpl. Schulbücher als Vokabelinstall. 39,- DM, D. Klaschka, 6334 Asslar PF 1227

PD-Soft, mit Disk, ST-Comp. 4. 2000er Serie, 2,30 DM/Stck., a.b. 16 h, Info: Tel. 02 01 / 73 34 97

Ausschreibung-Angebot-Rechnung
Datanorm/Baustoffverwaltung
K-Zahlberechnung-Wärmebedarf
Wärmeschutz-Dampfd./Tauwasser
Dipl. Ing. V. Koch, Germersweg 21
3563 Dautphetal 3, Tel. 0 64 68 / 76 52

Softstation
2 PD's von ST-Comp.
Fortl. Nr. (1-2 usw) DM 5,-
Bel. Komb. DM 7,-
sonstige Serien DM 7,-
Disketten wahlw. blau, weiß, rot, grün, gelb od. orange
Verp. Porto plus NN DM 7,-
Tel. 0 71 95 / 5 37 07

Anwender- u. Spielesoftware
Laufend n. Programme a. Lager
Riesenauswahl an Original
Soft- u. Hardware
Gratiskatalog anf. o. anrufen
W. Wünsch
Soft- u. Hardware,
0 72 31 / 76 65 95
Friedenstr. 212, 7530 Pforzheim

PD-Software
Je Disk 4,50 Porto/Verp./NN 6,-
SIGNUM-Pakete / PD-ABO.
KURZ-INFO kostenlos
bei: B. Jürgensen Hard & Soft
Holmberg 4 / 2398 Harrislee

Scanner Professional II, neu, OCR, Pegasus-Progr. etc, bis 600x 600 dpi, VB 2350,- 02 02 / 73 58 35

GFA-BASIC 3.5 EWS	238 DM
GFA-BASIC 3.0 EWS	168 DM
GFA-ASSEMBLER	128 DM
GFA-DRAFT Plus	328 DM
OMIKRON Basic-Compil.	169 DM
OMIKRON DRAW 3.0	129 DM
OMIKRON Turbo-Assembler	89 DM
OMIKRON Mortimer, d. Butler	79 DM

Über 1000 Public-Domain Disk. lieferbar!
Katalog-Diskette gegen 3,- DM in Bfm.
**** 24 Stunden Schnellversand ! ****
TITAN-DATA, Elisabethstr. 36, 4040 Neuss 21
Tel. 0 21 07 / 75 95 Mo-Fr 10 - 17 Uhr

G-I-R Software Vertrieb
Peter Grantz, Hauptstr. 49
D-2401 Ratekau/Lübeck

PD-Software für Atari ST,
- über 300 Disk.
Angebot wird lfd. erweitert.
- Update Service
- 24-Std.-Auftragsdienst gewährleistet durch
Tel.: 0 45 04 / 41 15
BTX: 0 45 04 / 52 67

Public Domain Disketten für Atari ST und Amiga in großer Auswahl und superbillig. Eine Liste gibt es gratis bei:
PD-Versand Anton Peter Maassen
Am Lindenplatz 17, 4040 Neuß

Der Hammer, Fraktale + Chaos f. ST
9*Mandelbrot, 10*Julia, 6*Biomorphe, 5*Attraktoren u.v.a., in GEM
200 k, nur SW, Unkostenbeitrag DM 50,-, bar o. Euroscheck
F. Szabo. A-1150, Sechshauserstr. 59

Komplette PD-Sammlung der ST-Computer Nr. 1-315 auf 205 Disk. 3.5" DM 999,- Tel.: 0 64 85 / 83 30

Grafiken u. Bilder für Signum u. STAD, ca 1500 Stck, inkl. Katalog und 3 Disketten DM 25,-
Tel.: 040 / 6 44 06 68

Signum-PD-Zeichensätze, 50 PD-Zeichensätze auf 2 Disketten, mit Katalog DM 25,-
Tel.: 040 / 6 44 06 68

ST-COMPUTER-PDs
SS 3DM/DS 4DM/P+V 5DM/NN+2DM
Katalogdisk 3DM bei: S.Heigert
Berndesallee 6, 6501 Heidesheim

GEBURTSHOROSKOPE!
Exakte Zeichnung, Aspekte, Berechnungsdaten, Interpretation für gewerbl. Einsatz!
Tel. 0 61 51 / 2 37 20

Original Megamax Laser C mit Quelltext-Debugger Laser DB dt. Handbuch + 2 dt. C-Lehrb. zu DM 200,-
Weiß Th., Blütenstr. 16, 8 München 40

ST-Computer u. **"2000er PD's"**
Jede Disk nur **3 DM. Signum PD-Fonts, Grafiken** für Signum/STAD u.a.
Katalog gegen 2 DM Rückporto bei:
S. Kraft, Albrechtstr. 1,
7920 Heidenheim

PD-SOFTWARE
Alle ST-Disks je DM 4,-
Versand: bei Scheck DM 3,-
bei Nachnahme DM 8,-
N. Twardoch, Wenkenstr. 67
4902 Bad Salzuflen

Wärmebedarf DIN 4701 • Wärmeschutz V • K-Zahl • Rohrnetz & Heizkörperauslegung • Demodisk DM 10,- Vorkasse • von J. Binder Behrensstr. 16, 5030 Hürth

---

Graphiksätze für ST von 1500 bis zum Jahr 1972. Info bei P. Sollfrank
PF 1121, 8419 Nittendorf

Public-Domain-Software aus der ST-Computer, Pool und GFA-Club je 3,40 DM inkl. 2S/2D-Disk
Lieferung innerhalb 24 Stunden!
Liste auf Disk lieferbar
Tel.: 040 / 7 12 54 97

Österreich: Atari ST Software
P.L. Payerg. 7/25, A-1160 Wien,
02 22 / 420-99-12 ab 18 h

PUBLIC-DOMAIN-SOFTWARE
Für PD-Freaks und Anwender.
Bei uns möglich: Zusammenstellen nach KByte, Kopie auf unsere /Ihre, single/double, Marken-/NN-Disk, Tausch u.v.m.
Große Auswahl! Gratiskatalog bei: Olaf Schwede, Röntgenweg 9/1, 7050 Waiblingen

Suche Software

Suche **CAMPUS-Draft** (Original!)
Tel.: 08 21 / 48 86 76

Biete Hardware

```
**************************************
*        RAM-Erweiterungen           *
*           spottbillig              *
**************************************
*  1040 ST/F auf 2 MB      348,-     *
*  520er auf 1 MB          175,-     *
*  520er auf 2,5 MB        475,-     *
*    • alle Erweit. inkl. Einbau •   *
*            Sven Betz               *
*    2 Hamburg 20 Hohe Weide 50      *
*       Tel. 404 / 420 43 63         *
**************************************
```

2-fach doppelseitiges 3.5" Disketten-Laufwerk für Atari ST
Tel.: 07 21 / 40 22 96

Speichererweiterung auf 2,5/ 4 MB 550,-/1100,- DM inkl. Einbau + Versand. 02 34 / 51 04 73

Scanner HAWK COLIBRI 400 dpi mit **Texterkennung** AUGUR
1000,- Dm Tel.: 02 02 / 40 04 99

Dataphon 2400 B mit Kabel und Software DM 550,-
Tel.: 08 41 / 7 15 84 ab 18 Uhr

Verk. 1040 STF o. Maus, 0431/569216

Atari-Rainbow-ROM-TOS 1.4(6.4.89) mit IBM-Grafikzeichen anstelle des hebräischen Alphabets + Patches aus ST-Computer 1/90 + Einbauanleitung, 90 DM,02630/7525

Verschenke: MEGA ST 2, SM 124, Maus, an den 90., der mir 20 DM schickt: an Thomas Gröning, Langeloh 4, 4600 Dortmund 50

Rainbow-TOS 1.4 DM 55,-
Doppel-TOS (1.4 + 1.0) **DM 79,-**
* mit Patches von ST-Computer 1/90 und Einbauanleitung *
Tel.: 02 12 / 20 86 13

Verkaufe Monitor Atari SM 124 für 230,- DM, Tel. 0 27 21 / 24 32

Speichererweiterung ST 260/520 auf 2,5 MB: DM 490,-/ 560,- DM auf 4 MB: DM 980,-/1120,- DM
07 21 / 37 66 22 excl./incl. Einbau

############LA 4825############
Black-Box-Logikanalysator mit 48 Kanal bei 25 MHz ab 2470,90 DM incl. Für PC's, Atari, MC 68000. Disassembler nachrüstbar,
bei: C.P.H. Tel. 0 74 23/60 66 Fax 60 67
############LA 4825############

Umständehalber zu verkaufen:
MEGA ST 2 (12/89) - NEC P6 - Syquest Wechselplatte 44 MB mit herausgeführtem SCSI- und DMA-Bus (12/89)
Tel.: 02 21 / 23 65 06 tgl. ab 17 h

Atari ST Uhrmodul (neuwertig)
VHB 60,- DM, Tel.: 0 61 51 / 14 42 35

Verk.: Mega ST2 + 2 (4 MB): 1950 DM dazu (eingebaut) Eickmann Turbo 16 (16 MHz) + Cache + ROM-TOS 1.4: 650 DM, VORTEX HD Plus 60: 990 DM, Ganzseitenbildschirm Matrix M 110 + Karte: 2900 DM, EPSON GT 4000 (S/W & Farbscanner, 400 dpi, 256 Graustufen, 16 Mio. Farben) + ST-Interface + TmS-Cranach: 3300 DM.
Gesamtanlage: 9100 DM.
Tel.: 0 64 21 / 2 23 10

I/O-INTERFACE ITF-32 TTL-komp.
32 Ein- u. Ausgänge, erweiterbar f. Alarmanlage, Ind.-Steuerungen
J. Sieber, Raffeltersteige 15
7100 Heilbronn, T. 0 71 31 / 16 11 51

I/O-Interface-Technik inkl. Treibersoftware u. Anwendungsschaltungen für ATARI
Info: K. Kellner, Adalbert-Stifter-Str. 12
3558 Frankenberg, Tel: 0 04 51 / 2 34 00

ST-225, 20 MB, 300 Fr., ATARI
SH-205, 20 MB, 400 Fr., ATARI
Color Monitor **SC 1224**, 400 Fr.,
HP41C + Memory + Kartenles. 300 Fr.
Tel.: 057 / 31 84 65

Super-Sonderangebot!!
1040 ST m. Uhr+Maus+SM 124+SH205 + 2. FDD **DM 1700,-**, 0 79 51 / 4 26 55

MODEME: 2400 Baud extern **333,-**
2400+ (+1200/75) **399,-**
2400 MNP 5 nur 555,-; **NEU:**
2400 MNP 5+ (+1200/75) **666,-**
BTX-Manager V.3.02 nur **255,-**
Info. anf. o. gleich bestellen:
Tel.: **0 64 22 / 34 38** CSR, 3575
Kirchhain, Breslauer Str. 19

Verschiedenes

Verkaufe Fußball-Toto-Prg. für 520 ST (Atari) auf Farbfernseher (selbstgeschrieben) Preis 40,- DM in Brief (per Einschreiben)
Herbert Schmidt, 3062 Bückeburg
Robert-Koch-Str. 6

TOS 1.4 ST-Computer Patch ohne Fastload auf 2 ROMs 95,- DM, neue Atari-Maus 30 DM, 069 / 5 07 16 94

TOS 1.4 nach ST-Computer 1/90 gepatcht 80,- DM, 0 21 04 / 3 50 83

MIDI Soft u. Hardware, 069 / 89 76 63

Biete ST-Computer Jahrg. '88/89
ST-Magazin '89, 40,- DM/Jahrg.
Blitter-ROM-TOS 30,- DM
Tel.: 0 81 67 / 86 71 ab 18 Uhr

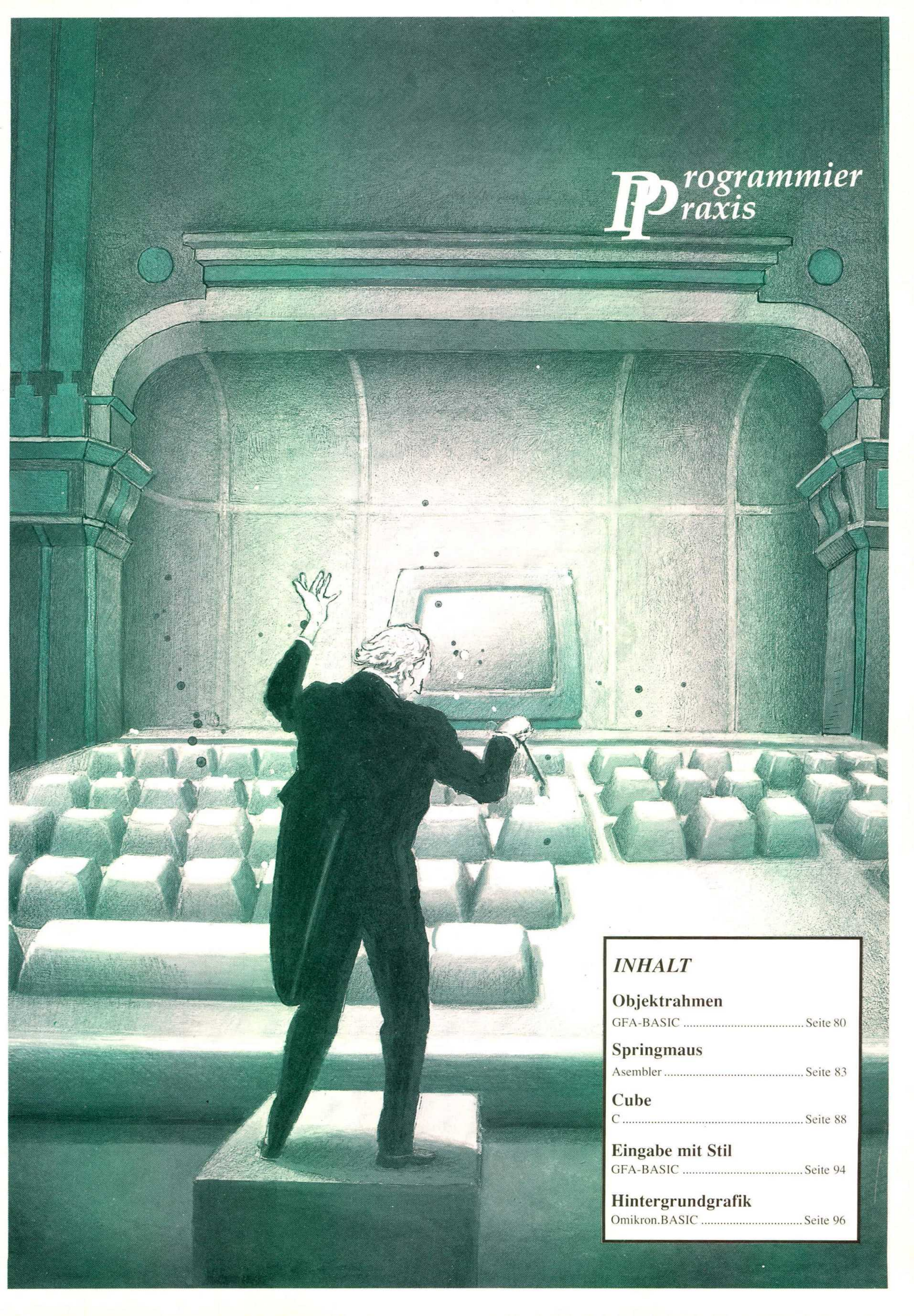
• • • MIDI e.V. • • •
Programmverleih, Zeitung
MIDI-PD's, Sounds + Samples
Hotline Tel.: 0 30 / 21 55 194
Mailbox Tel.: 0 30 / 78 15 103
MIDI e.V., PF 44 03 08, Berlin 44

Scanner DIN-A4 Vorlagen bis 600 dpi im STAD, GEM, IMG-Format.
Konvertierung nach HPGL mögl.
DM 2,-/Seite, Tel. 0 81 39 / 64 44

Das gr. GFA-Basic-Buch	DM 30
Atari ST Peeks + Pokes	DM 20
Video-Farb-Digitizer	DM 250

Alles neuwertig, wenig benutzt
Tel.: 0 85 31 / 75 23 ab 18 Uhr

Profiumbau 520ST in 19" Gehäuse
NT 50W, SCSI 48MB. MFLW 1.44 MB
4 MB RAM, abg. Tast. DM 3300,-
heiler engineering 07 21 / 37 66 22



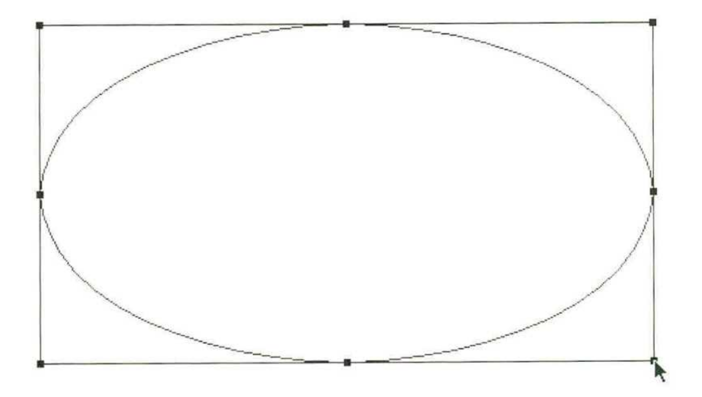
INHALT

Objektrahmen in GFA-BASIC 3.0

Dirk Beyer

In Calamus lassen sich Objektrahmen mit z.B. einer Ellipse aufspannen. Diese besitzen Grösseneinstellungsknöpfe und lassen sich verschieben.

Das Programm zeichnet einen Objektrahmen, der eine Ellipse beinhaltet. Auf dem Objektrahmenrand sitzen acht Größeneinstellungsknöpfe, die Sie mit der linken Maustaste anklicken können. Außerdem läßt sich der Rahmen verschieben. Das Feld *rahmen%()* beinhaltet die x-/y-Koordinaten der Einstellungsknopfmitten. Zur Abfrage, auf welchen Knopf geklickt wurde, wird eine Schleife durchlaufen, in der die Mauskoordinaten mit den Knopfflächen verglichen werden. Der evtl. gefundene Knopf hat den Feldindex *wahl_x%, wahl_y%*. Die Prozedur *rahmengroesse_ aendern* paßt dann der Mausbewegung die Koordinaten des angewählten Einstellungsknopfes an. Die neuen Koordinaten beinflussen auch andere Knöpfe, deren Koordinaten in der Prozedur *rahmenkoordinaten_zuweisen* aktualisiert werden. In der Prozedur *rahmen_zeichnen* wird an markierter Stelle das Objekt im Rahmen gezeichnet. Für eigene Objekte können Sie die Rahmenkoordinaten *x_links%,y_links%, x_rechts%,y_rechts%* (beziehen sich auf die linke obere und rechte untere Rahmenecke) und *x_mitte%, y_mitte%* verwenden. Das Verschieben des Rahmens erklärt der Kommentar in der Prozedur *rahmen_verändern*.



```
 1:  ' Objektrahmen in GFA-BASIC 3.0
 2:  ' Autor: Dirk Beyer
 3:  ' (c) MAXON Computer GmbH 1990
 4:  '
 5:  @vorarbeiten
 6:  @rahmen_zeichnen
 7:  @rahmen_veraendern
 8:  '
 9:  PROCEDURE vorarbeiten
10:    mindestbreite%=24  ! in Pixeln,Mindestmaße für
11:    mindesthoehe%=16   ! Objektrahmen
12:    PRINT AT(1,1);"Abbruch mit rechter Maustaste!"
13:    DIM rahmen%(5,5)
14:    x0%=200        !  Anfangskoordinaten des
                         Objektrahmens
15:    y0%=100        !  festlegen
16:    x1%=440
17:    y1%=300
18:    rahmen%(0,0)=x0%   ! X-Position des Rahmens
                            links
19:    rahmen%(0,1)=y0%   ! Y-Position oben
20:    rahmen%(0,4)=x1%   ! X-Position rechts
21:    rahmen%(2,1)=y1%   ! Y-Position unten
22:    rahmenkoordinaten_zuweisen
23:  RETURN
24:  PROCEDURE rahmen_veraendern
25:    REPEAT
26:      REPEAT
27:        MOUSE mx%,my%,mk%
28:      UNTIL mk%<>0  ! auf Maustaste warten
29:      IF mk%=1      ! falls linke Maustaste
                         gedrückt...
30:        groesse_aendern!=FALSE
31:        FOR x%=0 TO 4 STEP 2
32:          FOR y%=0 TO 2
33:            IF mx%>=rahmen%(y%,x%)-4 AND
                 mx%<=rahmen%(y%,x%)+4
34:              IF my%>=rahmen%(y%,x%+1)-4 AND
                   my%<=rahmen%(y%,x%+1)+4
35:                wahl_x%=x%    ! Wahl der
                                   Größeneinstellbox
36:                wahl_y%=y%
37:                SPUT screen$
38:                groesse_aendern!=TRUE
39:              ENDIF
40:            ENDIF
41:          NEXT y%
42:        NEXT x%
43:        IF groesse_aendern!=TRUE   ! Falls
                                        Größenbox
44:          rahmengroesse_aendern    ! angeklickt,
                                        Größe ändern
45:        ELSE                       ! sonst ...
46:          x0%=rahmen%(0,0)
```

→



```
 47:        y0%=rahmen%(0,1)
 48:        breite%=rahmen%(0,4)-x0%
 49:        hoehe%=rahmen%(2,5)-y0%
 50:        IF mx%>x0% AND mx%<x0%+breite%  ! falls
                                      Maus innerhalb des
 51:          IF my%>y0% AND my%<y0%+hoehe%
                        ! Rahmens: Rahmen verschieben
 52:            DEFMOUSE 4    ! Mauspfeil wird Hand
 53:            SPUT screen$
 54:            GRAPHMODE 3
 55:            BOX x0%,y0%,x0%+breite%,y0%+hoehe%
                         ! Move-Box zeichnen
 56:            no_mouse
 57:            mx1%=mx%    ! Mauskoordinaten merken
 58:            my1%=my%
 59:            REPEAT
 60:              MOUSE mx%,my%,mk%
 61:              IF mx%<>mx1% OR my%<>my1%
                         ! neue Mausposition ?
 62:                BOX x0%,y0%,x0%+breite%,y0%+
                      hoehe%  !alte Bbox löschen
 63:                x0%=MAX(0,MIN(x0%+mx%-mx1%,639-
                      breite%))
 64:                y0%=MAX(0,MIN(y0%+my%-my1%,399-
                      hoehe%))    ! und an
 65:                BOX x0%,y0%,x0%+breite%,y0%+
                   hoehe% ! neue Stelle zeichnen
 66:                mx1%=x0%+breite%/2
 67:                my1%=y0%+hoehe%/2
 68:              ENDIF
 69:            UNTIL mk%<>0
 70:            BOX x0%,y0%,x0%+breite%,y0%+hoehe%
                         ! Move-Box löschen
 71:            rahmen%(0,0)=x0% ! Rahmenkoordinaten
                                  aktualisieren
 72:            rahmen%(0,1)=y0%
 73:            rahmen%(0,4)=x0%+breite%
 74:            rahmen%(2,1)=y0%+hoehe%
 75:            @rahmenkoordinaten_zuweisen
 76:            GRAPHMODE 1
 77:            rahmen_zeichnen  ! Rahmen mit Objekt
                              an neue Stelle zeichnen
 78:            no_mouse
 79:            DEFMOUSE 0    ! wieder Pfeil als Maus
 80:          ENDIF
 81:        ENDIF
 82:      ENDIF
 83:    ENDIF
 84:  UNTIL mk%=2  ! Ende nach Drücken der rechten
                 Maustaste
 85:  EDIT
 86: RETURN
 87: PROCEDURE no_mouse
 88:  REPEAT
 89:  UNTIL MOUSEK=0   ! warten, bis kein Mausklick
 90: RETURN
 91: PROCEDURE rahmengroesse_aendern
                    !Objektrahmengröße ändern
 92:  DEFMOUSE 7       ! Mauspfeil wird Kreuz
 93:  GRAPHMODE 3
 94:  rahmen_zeichnen
 95:  mx%=rahmen%(wahl_y%,wahl_x%)     ! mx%,my% wird
                                      die Mitte der
 96:  my%=rahmen%(wahl_y%,wahl_x%+1) ! gewählten
                                   Größeneinstell-
 97:  '                                box zugewiesen
 98:  SETMOUSE mx%,my%          ! Mauspfeil in die
                                 Mitte der gewählten
 99:  '                  Größeneinstellungsbox setzen
100:  mx1%=mx%
101:  my1%=my%
102:  no_mouse
103:  REPEAT
104:    MOUSE mx%,my%,mk%
105:    IF mx%<>mx1% OR my%<>my1%
106:      mx1%=mx%
107:      my1%=my%
108:      alt_x0%=rahmen%(0,0)           ! bisherige
                                 Rahmenkoordinaten
109:      alt_y0%=rahmen%(0,1)          ! an "alt"-
                                   Variablen zuweisen
110:      alt_x1%=rahmen%(2,4)
111:      alt_y1%=rahmen%(2,5)
112:      IF wahl_x%=0 OR wahl_x%=4
                   ! Rahmenkoordinaten entsprechend
113:        rahmen%(0,wahl_x%)=mx%
                  ! der Mausverschiebung ändern,
114:        rahmen%(1,wahl_x%)=mx%
                     ! d.h Rahmengröße wird verändert
115:        rahmen%(2,wahl_x%)=mx%
116:      ENDIF
117:      rahmen%(wahl_y%,1)=my%
118:      rahmen%(wahl_y%,3)=my%
119:      rahmen%(wahl_y%,5)=my%
120:      rahmenkoordinaten_zuweisen
121:      neu_x0%=rahmen%(0,0)
                          ! Neue Rahmenkoordinaten
122:      neu_y0%=rahmen%(0,1)
123:      neu_x1%=rahmen%(2,4)
124:      neu_y1%=rahmen%(2,1)
125:      IF neu_x1%-neu_x0%<mindestbreite%
                   ! prüfen, ob neuer Rahmen
126:        IF neu_x1%<>alt_x1%
                   ! Mindestmaße hat, wenn nicht...
127:          rahmen%(0,4)=neu_x0%+mindestbreite%
                   ! GröPe= Mindestmaße
128:          rahmen%(1,4)=rahmen%(0,4)
129:          rahmen%(2,4)=rahmen%(0,4)
130:        ELSE
131:          rahmen%(0,0)=rahmen%(0,4)-
                        mindestbreite%
132:          rahmen%(1,0)=rahmen%(0,0)
133:          rahmen%(2,0)=rahmen%(0,0)
134:        ENDIF
135:        rahmenkoordinaten_zuweisen
136:      ENDIF
137:      IF neu_y1%-neu_y0%<mindesthoehe%
138:        IF neu_y1%<>alt_y1%
139:          rahmen%(2,1)=rahmen%(0,1)+mindesthoehe%
140:          rahmen%(2,3)=rahmen%(0,5)
141:          rahmen%(2,5)=rahmen%(0,5)
142:        ELSE
143:          rahmen%(0,1)=rahmen%(2,1)-mindesthoehe%
144:          rahmen%(0,3)=rahmen%(0,1)
145:          rahmen%(0,5)=rahmen%(0,1)
146:        ENDIF
147:        rahmenkoordinaten_zuweisen
148:      ENDIF
149:      neu_x0%=rahmen%(0,0)
150:      neu_y0%=rahmen%(0,1)
151:      neu_x1%=rahmen%(2,4)
152:      neu_y1%=rahmen%(2,1)
153:      ' nur dann alten Rahmen löschen und neuen
          zeichnen, wenn
154:      ' neue Koordinaten sich von bisherigen
          (alten) unterscheiden
155:      IF neu_x0%<>alt_x0% OR neu_x1%<>alt_x1%
156:        BOX alt_x0%,alt_y0%,alt_x1%,alt_y1%
157:        rahmen_zeichnen
158:      ELSE
159:        IF neu_y0%<>alt_y0% OR neu_y1%<>alt_y1%
160:          BOX alt_x0%,alt_y0%,alt_x1%,alt_y1%
161:          rahmen_zeichnen
162:        ENDIF
163:      ENDIF
164:    ENDIF
165:  UNTIL mk%<>0
166:  rahmen_zeichnen
167:  groesse_aendern!=FALSE
168:  GRAPHMODE 1
169:  SGET screen$
170:  rahmen_zeichnen
171:  REPEAT
172:  UNTIL MOUSEK=0
173:  DEFMOUSE 0
174: RETURN
175: PROCEDURE rahmen_zeichnen   ! Objektrahmen+Objekt
                                     zeichnen
176:  IF NOT groesse_aendern!
177:    SGET screen$  ! Hintergrund retten
178:  ENDIF
179:  ' Rahmenbox zeichnen...
180:  BOX rahmen%(0,0),rahmen%(0,1),rahmen%(2,4),
      rahmen%(2,5)
181:  x_links%=rahmen%(0,0)
182:  x_mitte%=rahmen%(0,2)
183:  x_rechts%=rahmen%(0,4)
184:  y_oben%=rahmen%(0,1)
185:  y_mitte%=rahmen%(1,1)
186:  y_unten%=rahmen%(2,1)
187:  IF NOT groesse_aendern!
188:    ' Größreneinstellungsboxen zeichnen...
189:    PBOX x_links%-2,y_oben%-2,
          x_links%+2,y_oben%+2
```

→

```
190:      PBOX x_links%-2,y_mitte%-2,
               x_links%+2,y_mitte%+2
191:      PBOX x_links%-2,y_unten%-2,
               x_links%+2,y_unten%+2
192:      PBOX x_rechts%-2,y_oben%-2,
               x_rechts%+2,y_oben%+2
193:      PBOX x_rechts%-2,y_mitte%-2,
               x_rechts%+2,y_mitte%+2
194:      PBOX x_rechts%-2,y_unten%-2,
               x_rechts%+2,y_unten%+2
195:      PBOX x_mitte%-2,y_oben%-2,
               x_mitte%+2,y_oben%+2
196:      PBOX x_mitte%-2,y_unten%-2,
               x_mitte%+2,y_unten%+2
197:      ' Hier können Sie Ihr Rahmenobjekt einfügen:
198:      ' Rahmenobjekt zeichnen...
199:      ELLIPSE x_mitte%,y_mitte%,x_rechts%-x_mitte%,
                  y_unten%-y_mitte%
200:    ENDIF
201:  RETURN
202:  PROCEDURE rahmenkoordinaten_zuweisen
203:    ' rahmen%() enthält Koordinaten der
          Größeneinstellungsboxen
204:    rahmen%(0,2)=(rahmen%(0,0)+rahmen%(0,4))/2
205:    rahmen%(0,3)=rahmen%(0,1)
206:    rahmen%(0,5)=rahmen%(0,1)
207:    rahmen%(1,0)=rahmen%(0,0)
208:    rahmen%(1,1)=(rahmen%(0,1)+rahmen%(2,1))/2
209:    rahmen%(1,4)=rahmen%(0,4)
210:    rahmen%(1,5)=rahmen%(1,1)
211:    rahmen%(2,0)=rahmen%(0,0)
212:    rahmen%(2,2)=rahmen%(0,2)
213:    rahmen%(2,3)=rahmen%(2,1)
214:    rahmen%(2,4)=rahmen%(0,4)
215:    rahmen%(2,5)=rahmen%(2,1)
216:  RETURN
```

SPRINGMAUS

Ralf Stachs

Klappen Ihnen auch andauernd unbeabsichtigt Menüleisten herunter? Springmaus verhindert diese lästigen Unruhestiftereien. Wie das geht? Lesen Sie selbst...

Springmaus verhindert das unbeabsichtigte Herunterklappen der Menüleiste, was z.B. bei einem Ausdruck von Dateien mit 1st_Word den Druckvorgang unterbricht. Installiert wird Springmaus aus dem AUTO-Ordner oder vom Desktop. Nach der Installation kann sich der Mauszeiger nur im Bereich unterhalb der Menüleiste bewegen. Das Herunterklappen der Menüleiste geschieht jetzt mit der sonst ungenutzten rechten Maustaste. Durch das Drücken der rechten Maustaste springt der Mauszeiger in die Menüleiste und bleibt dort stehen, bis die Taste wieder losgelassen wird.

Durch einen erneuten Start wird Springmaus ausgeschaltet, was durch eine Textausgabe bestätigt wird. Der Mauszeiger verhält sich jetzt so, als sei Springmaus nicht installiert worden. Ein nochmaliger Start von Springmaus schaltet das Programm wieder ein, was auch durch eine Textausgabe bestätigt wird.

Im File-Namen können verschiedene Werte eingestellt werden. Der siebte Buchstabe des File-Namens gibt an, ob nach einem Sprung in die Menüleiste zur alten Position des Mauszeigers oder ins Pull Down-Menü gesprungen wird. Dabei bedeutet 'A', daß zur alten Position des Mauszeigers gesprungen wird. Werte von '1' bis '9' geben an, daß ins Pull Down-Menü gesprungen wird. Für den Sprung zu dem ersten Eintrag in der Menüleiste steht '1'.

Der achte Buchstabe gibt an, ob die rechte Maustaste an die original Mausroutine weitergegeben wird. Das Ausblenden der rechten Maustaste ist für den Fall, daß ein anderes Programm auch die rechte Maustaste für eine Funktion benutzt, und diese Funktion bei jedem Herunterklappen der Menüleiste ausgelöst wird. Wenn die rechte Maustaste ausgeschaltet werden soll, muß der achte Buchstabe des File-Namens ein 'A' sein. Bei einem 'E' als achtem Buchstaben ist die rechte Maustaste eingeschaltet.

z B.:

SMAUS_xy.PRG

x	
A	Springe zur alten Position
1	Springe zum ersten Eintrag in Menüleiste (Mono/Farbe)
2	Springe zum zweiten Eintrag in Menüleiste (Mono/Farbe)
3	Springe zum dritten Eintrag in Menüleiste (Mono/Farbe)
4	Springe zum vierten Eintrag in Menüleiste (Mono/Farbe)
5	Springe zum fünften Eintrag in Menüleiste (Mono/Farbe)
6	Springe zum sechsten Eintrag in Menüleiste (Mono/Farbe)
7	Springe zum siebten Eintrag in Menüleiste (Mono/Farbe)
8	Springe zum achten Eintrag in Menüleiste (Farbe)
9	Springe zum neunten Eintrag in Menüleiste (Farbe)

y	
A	rechte Maustaste ausgeschaltet
E	rechte Maustaste eingeschaltet

Tabelle 1: So muß der Dateiname ergänzt werden

Filename 'SMAUS_AA.PRG'
Es erfolgt ein Sprung zur alten Position des Mauszeigers. Die rechte Maustaste ist ausgeschaltet.

Filename 'SMAUS_2E.PRG'
Es wird zum zweiten Eintrag in der Menüleiste gesprungen. Die rechte Maustaste ist eigeschaltet.
TOS, Ganzeitenmonitor, Farbe und monochrom

Springmaus läuft unter TOS von 1986 sowie im Farbmodus und mit Monochrommonitor. Vielleicht läuft Springmaus auch mit Blitter-TOS und dem neuen TOS 1.4, was ich leider nicht testen konnte. Auch das Testen mit einem Ganzseitenmonitor ist mir verwehrt geblieben.

Programmbeschreibung

Installation Desktop

Wird Springmaus aus dem Desktop gestartet, überprüft das programm, ob es schon installiert ist. Wenn dies der Fall ist, wird Springmaus ein- oder ausgeschaltet. Falls Springmaus nicht installiert ist, wird die Adresse des Mausvektors über die XBIOS-Routine Kbdvbase geholt. Der original Mausvektor wird gespeichert. Danach leitet man den Mausvektor im Supervisor-

modus auf die neue Mausroutine um.

Installation AUTO-Ordner
Wenn sich keine GEM-Applikation mit der AES-Funktion *APPL_INIT* anmelden läßt, wurde Springmaus aus dem AUTO-Ordner gestartet. Zuerst wird der AUTO-Ordner geöffnet, danach ein VBL-Slot belegt und der Mausvektor mit einem Merker belegt.

VBL-Slot
Die VBL-Routine überprüft, ob der Merker noch gesetzt ist. Wenn dies nach Initialisierung des Mausvektors durch GEM nicht der Fall ist, wird der alte Mausvektor gespeichert und der Mausvektor auf die neue Mausroutine umgeleitet.

Parameter
Mit *FSFIRST* und der Suchmaske 'SMAUS_??.PRG' holt man die Parameter.

Neue Mausroutine
Die neue Mausroutine testet, ob Springmaus ein- oder ausgeschaltet ist. Danach prüft man, ob ein Mauspaket angekommen ist. Dann wird der neue *GCURY*-Wert berechnet und die original Mausroutine angesprungen.

Springen
Bei Drücken der rechten Maustaste wird die original Mausroutine solange aufgerufen, bis der Mauszeiger in der Menüleiste ist. Nach dem Loslassen der Maustaste erfolgt der Sprung zur alten Position oder ins Pull Down-Menü.

Negative Line-A-Variablen
Von den negativen Line-A-Variablen (ATARI ST Profibuch, Sybex-Verlag) werden *GCURY (-$258)*, der die absolute y-Position der Maus enthält, sowie *MOUSE_FLAG ($153)*, die anzeigt, ob der Maus-Interrupt ein- oder ausgeschaltet ist, benutzt.



```
 1:  *21.8.89-----------------*
 2:  *  Spring Maus           *
 3:  *  8/89 Ralf Stachs      *
 4:  *------------------------*
 5:  *                        *
 6:  *   Maxon Computer GmbH  *
 7:  *                        *
 8:  *------------------------*
 9:
10:  *TRAPS
11:  GEMDOS  equ 1
12:  BIOS     equ 13
13:  XBIOS    equ 14
14:
15:  *Daten der neuen Mausroutine
16:  resi_anfang:
17:  bra INSTALL
18:                  dc.b 0        *Füller
19:  maus_aus:       dc.b 0        *schaltet Spring
                                    Maus an/aus
20:  dc.b "XBRA"                   *XBRA Protokoll
21:  dc.b "RS_1"                   *eigene Kennung
22:  old_vec: dc.l 0               *alter Vektor
23:
24:                  *neue Mausroutine
25:  new_mouse:      *neue Routine eingeschaltet
26:                  tst.b maus_aus  *neue Routine
                                      ausgeschaltet
27:                  bne aus        *ja
28:
29:                  movem.l d0-d1/a0-a3,-(sp)
                                    *Register retten
30:
31:                  *Maus Header ?
32:                  move.b (a0),d0 *Header Kopf
                                    in d0
33:                  cmp.b #$f8,d0   *kleiner $f8
34:                  blt end         *ja
35:                  cmp.b #$fb,d0   *größer $fb
36:                  bgt end         *ja
37:
38:                  *Maus aus
39:                  move.l tos_aus,a1
                     *Flag für Maus aus (MOUSE_FLAG)
40:                  tst.b (a1)      *Maus aus
41:                  bne end         *ja dann Ende
42:
43:                  *neues GCURY berechnen
44:                  move.b 2(a0),d1 *Wert header_y
                                      nach d1
45:                  move.l A_GCURY,a2
                     *Adresse A_GCURY nach a2
46:                  move.w (a2),d0  *Wert GCURY
                                      nach d0
47:                  ext.w d1        *header_y Vor-
                               zeichenrichtig auf .w
48:                  add.w d1,d0     *header_y zu
                                     A_GCURY addieren
49:
50:                  *Maus an min. Grenze
51:                  cmp.w min,d0    *GCURY <min
                                      (Menüleiste)
52:                  bgt sp_6        *nein
53:                  tst.w (a2)      *GCURY = 0
54:                  bne sp_5        *nein
55:                  move.b #0,2(a0) *k. y Bewegung
56:                  bra sp_6        *Taste abfragen
57:
58:                  *Mausbewegung bis min. Wert
59:  sp_5:           move.b min+1,d1 *min. Wert
                                      GCURY nach d1
60:                  move.b 1(a2),d0 *alter GCURY
                                      Wert nach d0
61:                  sub.b d0,d1     *neue Mausbewe-
                                      gung nach oben
62:                  move.b d1,2(a0) *header_y noch
                                      bis min. Wert
63:
64:                  *rechte Maustaste ?
65:  sp_6:           lea r_taste,a1  *Adr. r_taste
                                      nach a1
66:                  cmp.b #$f9,(a0) *rechte Mausta-
                                      ste gedrückt
67:                  bne sp_3        *nein
68:
69:                  *rechte Taste weitergeben
70:                  tst.b r_weiter  *rechte Taste
                                      weitergeben
71:                  bne sp_2        *weitergeben
72:                  and.b #%11111110,(a0)
                     *Bit rechte Maustaste löschen
73:
74:  sp_2:           tst.b (a1)      *rechte Taste
75:                  *beim letzten mal gedrückt
76:                  bne end         *ja
77:
78:                  *Maus nach oben
79:                  move.b #1,r_taste
                     *Flag setzen Taste gedrückt
80:                  move.w (a2),merk_y
                     *GCURY speichern
81:                  and.b #%11111110,(a0)
                     *Bit rechte Maustaste löschen
82:  sp_7:           move.l old_vec,a3
                     *Adresse Mausroutine
83:                  move.b #-127,2(a0)
                     *Maus nach oben
84:                  jsr (a3)        *alte Routine
85:                  move.l A_GCURY,a3
                     *Adresse von GCURY
86:                  tst.w (a3)      *GCURY gleich 0
87:                  beq sp_10       *ende
88:                  bra sp_7        *nochmal
89:
90:  sp_3:           *Maus in Pull Down
91:                  tst.b (a1)      *rechte Taste
                                      letztes Mal
92:                                  *nicht gedrückt
93:                  beq end         *ja
94:                  move.b #0,(a1)  *Flag löschen
95:                  tst.b spring    *spring zu alter
                                      Position
```



```
 96:               bne sp_9         *ja
 97:               move.b pull_down,2(a0)
                   *Maus in Pull Down
 98:               bra end          *alte Position
 99:                                *überspringen
100:  sp_9:        *Maus alte Position
101:               lea merk_y,a1    *Adresse von
                                     merk_y nach a1
102:               move.b 1(a1),2(a0)
                   *Wert beim letztem Aufruf
103:               move.w #127,d0   *max. Wert in d0
104:               sub.w d0,(a1)    *altes GCURY-127
105:               bmi end          *k. Aufruf mehr
                                    GCURY <127
106:               move.l old_vec,a3
                   *Adresse Mausroutine
107:               move.b #127,2(a0)
                   *max. Wert in Header schreiben
108:               jsr (a3)         *alte Routine
109:               bra sp_9
110:
111:               *rechte Taste weitergeben ?
112:  sp_10:       tst.b r_weiter   *rechte Taste
                                     weitergeben
113:               beq sp_11        *n. weitergeben
114:               or.b #%00000001,(a0)
                   *Bit rechte Maustaste setzen
115:  sp_11:       move.b #0,1(a0) *Maus n. bewegen
116:               move.b #0,2(a0) *Maus n. bewegen
117:               move.l old_vec,a3
                   *Adresse Mausroutine
118:               jsr (a3)         *alte Routine
119:
120:  end:         *alte Mausroutine anspringen
121:               movem.l (sp)+,d0-d1/a0-a3
                   *Register zurückschreiben
122:  aus:         move.l old_vec,-(sp)
                   *alte Mausroutine
123:               rts              *anspringen
124:
125:               *VBL-Routine bei Start aus
                    AUTO-Ordner
126:  neu_vbl: move.l kbdv_addr,a0      *kdbv_addr->a0
127:               lea 16(a0),a1    *Adr. Vektor a1
128:               cmp.l #new_mouse,(a1)
                   *Merker noch da
129:               beq sp_20        *ja
130:
131:               *vbl-Slot löschen
132:               move.l vbl_slot,a0
133:               move.l #0,(a0)
134:
135:               move.l (a1),old_vec
                   *alten Mausvektor sichern
136:               move.l #new_mouse,(a1)
                   *neue Mausroutine init
137:  sp_20:       rts
138:
139:
140:  *Flags
141:  *r_taste     0=rechte Taste nicht gedrückt
142:  *            1=rechte Taste gedrückt
143:  *r_weiter    0=rechte Taste nicht weitergeben
               an Origi. Mausroutine

144:  *            1=rechte Taste weitergeben an
               Origi.Mausroutine

145:  *maus_aus    0=neue Mausroutine eingeschaltet
146:  *            1=neue Mausroutine ausschalten
147:  *spring      0=spring in Pull Down
148:  *            1=spring zur alten Position
149:  r_taste:     dc.b 0
150:  r_weiter:    dc.b 0
151:  spring:      dc.b 1
152:
153:  *Variablen
154:  *merk_y  =merkt sich maus_y
155:  *min     =min. y Wert von maus_y  20-Mono
                11-Farbe
156:  *tos_aus =Adresse von Flag TOS
157:  *     0=Maus wird angezeigt
158:  *     1=Maus wird nicht angezeigt
159:  *kbdv_addr         =Adresse der kbdv Tabelle
160:  *old_vec =Zeiger auf Original Mausroutine
161:  *vbl_slot          =Adresse des VBL-Slot
162:  *A_GCURY =Adresse von GCURY
163:  *pull_down =Wert nach unten bei Pull Down
164:   EVEN
165:  merk_y:  dc.w 0
166:  tos_aus:          dc.l 0
167:  kbdv_addr:        dc.l 0
168:  vbl_slot:         dc.l 0
169:  A_GCURY: dc.l 0
170:  min:              dc.w 0
171:  pull_down:        dc.b 0
172:  resi_ende:
173:
174:   TEXT
175:                    *installierung von Spring Maus
176:                    *gesamt-PRG Speicher belegen
177:   EVEN
178:  INSTALL: move.l   sp,a6          *Adr. BASEPAGE
179:                    lea     USTACK,sp  *neuer Stack
180:
181:                    move.l  4(a6),a6
                        *Speicher belegen
182:                    move.l  $c(a6),a4
183:                    adda.l  $14(a6),a4
184:                    adda.l  $1c(a6),a4
185:
186:                    pea             256(a4)
187:                    pea             (a6)
188:                    clr.w   -(sp)
189:                    move.w  #74,-(sp)
190:                    *Mshrink aufrufen
191:                    trap #GEMDOS
192:                    lea 12(sp),sp
193:
194:                    *Adresse KBDVECS holen
195:                    move #34,-(a7) *Kbdvbase rufen
196:                    trap #XBIOS
197:                    addq.l #2,a7
198:                    move.l d0,kbdv_addr *Adresse
                           der Vektortabelle sichern
199:
200:                    *Spring Maus schon installiert
201:                    move.l kbdv_addr,a1 *Adresse
                           der Vektortabelle nach a1
202:                    move.l 16(a1),a2
                        *Adresse von mausvektor in a2
203:                    lea   (a2),a0
                        *Adresse der mausroutine
204:                    cmpi.l #"RS_1",-8(a0)
                        *vergleichen
205:                    bne sp_50          *Spring Maus
                                            installieren
206:
207:                    *Spring Maus ein oder
                                  ausschalten
208:                    not.b -13(a0)     *maus_aus
                                            invertieren
209:                    bne sp_55
210:                    pea string5       *Spring Maus
                                            einschalten
211:                    move.w  #9,-(sp)
212:                    trap #GEMDOS
213:                    addq.l #6,sp
214:                    bra sp_56
215:  sp_55:            pea string6       *Spring Maus
                                            auschalten
216:                    move.w  #9,-(sp)
217:                    trap #GEMDOS
218:                    addq.l #6,sp
219:                    *PRG beenden
220:  sp_56:            clr.w  -(sp)
221:                    trap #GEMDOS
222:
223:  sp_50:            *alten Vektor sichern
224:                    lea old_vec,a0
225:                    move.l a2,(a0)
226:
227:                    *AES-anmelden
228:                    move.w #10,d0     *Applikation
                                            anmelden
229:                    bsr aes_init      *AES aufrufen
230:                    tst.w aes_global
                        *starten aus AUTO-Ordner
231:                    beq sp_52         *j,k.ap_version
232:
233:                    *Vom Desktop starten
234:                    *maus installieren
```

→

```
235:              lea desktop,a0  *Adr.v. Desktop
                                   nach a0
236:              pea     (a0)
237:              move.w #38,-(sp)  *Supexec
238:              trap #XBIOS
239:              addq.l #6,sp
240:              bra sp_51
241:
242: sp_52:       *Aus AUTO-Ordner starten
243:              *AUTO Ordner öfnen
244:              pea path        *AUTO Ordner
245:              move.w #$3b,-(sp)    *Dsetpath
246:              trap #1
247:              addq.l #6,sp
248:
249:              *Unterprogramm im Supervisor-
                   Modus ausführen
250:              lea auto,a0     *Adresse von
                                   auto nach a0
251:              pea     (a0)
252:              move.w #$26,-(sp)  *Supexec
253:              trap #XBIOS
254:              addq.l #6,sp
255:
256: sp_51:       *Bildschirmauflösung einstellen
257:              *                Farbe |  Mono
258:              *min             11    |  20
259:              move.w #4,-(sp)*Getrez aufrufen
260:              trap #XBIOS
261:              addq.l #2,sp
262:
263:              *Farbe
264:              move.w #11,min  *Menüleiste
265:              cmp.b #1,d0     *Farbe
266:              ble sp_61       *ja
267:
268:              *Mono
269:              move.w #20,min  *Menüleiste
270:
271: sp_61:       *Filename holen
272:              pea buf         *Puffer DTA
273:              move.w #$1a,-(sp)
                  *Fsetdta aufrufen
274:              trap #GEMDOS
275:              addq.l #6,sp
276:              move.w #0,-(sp) *normale Datei
277:              pea     fspec   *Suchname
                                  (SMAUS_??.PRG)
278:              move.w #$4e,-(sp)  *Fsfirst
279:              trap #GEMDOS
280:              addq.l #8,sp
281:
282:              *0   ->alte Position
283:              *1-9 ->Pull Down
284:              cmp.b #"A",buf+36 *alte Pos.
285:              beq sp_53       *ja
286:
287:              *Wert für Pull Down berechnen
288:              move.b #0,spring   *Pull Down
289:              clr.l d0          *d0 löschen
290:              move.b buf+36,d0    *Faktor
291:              sub.w #"1",d0   *Faktor in d0
292:
293:              cmp.w #20,min   *Mono
294:              beq sp_62       *ja
295:
296:              *Farbe
297:              mulu.w #8,d0   *Wert f.PullDown
298:              add.w min,d0   *add Menüleiste
299:              add.w #5,d0    *und Zeilenmitte
300:              bra sp_63      *Mono überjumpen
301:
302: sp_62:       *Mono
303:              mulu.w #16,d0  *Wert f.PullDown
304:              add.w min,d0   *add Menüleiste
305:              add.w #10,d0   *und Zeilenmitte
306:
307: sp_63:       *max. und min prüfen
308:              cmp.w #127,d0  *kleiner als 127
309:              blt sp_60      *ja
310:              move.b #127,d0 *max. Wert
311: sp_60:       cmp.w min,d0   *größer als
                                 Menüleiste
312:              bgt sp_59      *ja
313:              move.w min,d0  *min. Wert
314: sp_59:       move.b d0,pull_down  *speichern
315:
316: sp_53:       *rechte Maus-Taste ein oder aus
317:              cmp.b #"A",buf+37 *r. Taste aus
318:              beq sp_54        *ja
319:              move.b #1,r_weiter*r. Taste an
320:
321: sp_54:       *tos_aus holen
322:              dc.w $a000       *Adr.neg.Line-A
323:              sub.l #$153,a0   *Adr.MAUSFLAG
324:              move.l a0,tos_aus *und merken
325:
326:              *Adresse von GCURY speichern
327:              move.l d0,a0     *Adr.neg.Line-A
328:              sub.l #$258,a0   *Adresse GCURY
329:              move.l a0,A_GCURY *und merken
330:
331:              *initialisierung i.o.
332: fertig:      pea string2
333:              move.w  #9,-(sp)
334:              trap #GEMDOS
335:              addq.l #6,sp
336:
337:              *Speicherplatz für Mausroutine
338:              clr.w -(sp)
339:              pea resi_ende-resi_anfang+256
340:              move.w #49,-(a7)    *Ptermres
341:              trap #GEMDOS
342:
343:              *PRG im Supervisor mausroutine
                   installieren
344: auto:        *starten aus AUTO Ordner
345:              *neuen Vektor setzen
346:              move.l kbdv_addr,a0
                  *Keyboardadresse nach a0
347:              move.l #new_mouse,16(a0)
348:              *Merker setzen
349:              *VBL installieren
350:              move.w $454,d0   *Anzahl VBL
                                   Routinen (nvbls)
351:              lsl #2,d0        *Anzahl*4
352:              move.l $456,a0   *Zeiger auf VBL
                                  (_vblqueue)
353:              clr d1           *Zähler
354: weiter:      tst.l 4(a0,d1)   *VBL Slot frei
355:              beq frei         *ja
356:              add #4,d1        *next Pointer
357:              cmp.l d0,d1      *alle abgefragt
358:              bne weiter
359:
360:              *Spring Maus nicht initialisieren
361:              pea string3      *Alle besetzt
362:              move.w  #9,-(sp)
363:              trap #GEMDOS
364:              addq.l #6,sp
365:              *PRG beenden
366:              clr.w  -(sp)
367:              trap #GEMDOS
368:
369: frei:        lea 4(a0,d1),a2 *Adr. Slot in a2
370:              lea neu_vbl,a1   *Adr.neue Rout.
371:              move.l a1,(a2)   *neue Routine
                                   einhängen
372:              lea vbl_slot,a1 *Slot Adresse
373:              move.l a2,(a1)   *sichern
374:              rts
375:
376: desktop: *Starten vom Desktop
377:              move.l kbdv_addr,a0
                  *Keyboardadresse nach a0
378:              lea new_mouse,a1
                  *Adresse neue Mausroutine
379:              move.l a1,16(a0)
                  *neuen Vektor setzen
380:              rts
381:
382: aes_init:    *AES aufrufen
383:              lea contrl,a0    *Adr.contrl->a0
384:              move.w d0,(a0)+ *Opcode
385:              clr.w (a0)+      *eintr. int_in
386:              move.w #1,(a0)   *eintr. int_out
387:              clr.w (a0)+      *eintr. addr_in
388:              clr.w (a0)       *eintr. addr_out
389:
390:              move.l #aes_data,d1
                  *Adresse AES-Array
391:              move.w #$c8,d0   *AES_Aufruf
392:              trap #2
393:              rts
```

→

```
394:
395:    DATA
396:   string2: dc.b 13,10,"+---------------------+"
397:            dc.b 13,10,"+    Spring MAUS 3.3    +"
398:            dc.b 13,10,"+    9/89 Ralf Stachs   +"
399:            dc.b 13,10,"+  (c) MAXON Computer +"
400:            dc.b 13,10,"+---------------------+"
401:            dc.b 13,10,"SMAUS_xy.PRG"
402:            dc.b 13,10,"-x"
403:            dc.b 13,10,"A  ->Alte Position"
404:            dc.b 13,10,"1-9->In Pull Down"
405:            dc.b 13,10,"-y"
406:            dc.b 13,10,"E->Rechte Maustaste ein"
407:            dc.b 13,10,"A->Rechte Maustaste aus",0
408:   string3: dc.b 13,10,"Alle VBL-Slots besetzt",0
409:   string5: dc.b 13,10,"SPRING MAUS EIN",0
410:   string6: dc.b 13,10,"SPRING MAUS AUS",0
411:   fspec:   dc.b "SMAUS_??.PRG",0
412:   path:    dc.b "\auto\",0
413:
```

```
414:    EVEN
415:   aes_data:        dc.l contrl
416:                    dc.l aes_global
417:                    dc.l init_in
418:                    dc.l init_out
419:                    dc.l addr_in
420:                    dc.l addr_out
421:
422:    BSS
423:   aes_global:      ds.w 15
424:   contrl:     ds.w 10
425:   init_in: ds.w 128
426:   init_out:        ds.w 128
427:   addr_in: ds.l 128
428:   addr_out:        ds.l 128
429:
430:   buf:     ds.b 100
431:            ds.b 256
432:   USTACK:  ds.w 0
```

CUBE - der drehbare Würfel

Justus Piater

ICH HABE EINE VORLIEBE FÜR ABSOLUT NUTZLOSE, ABER DAFÜR MATHEMATISCH INTERESSANTE PROGRAMMIERPROBLEME. CUBE IST EINES MEINER AUF DIESE WEISE ENTSTANDENEN PROGRAMME, DESSEN VERSTÄNDNIS ALLERDINGS EINIGES AN DREIDIMENSIONALER VORSTELLUNGSKRAFT SOWIE MATHEMATISCHE OBERSTUFENKENNTNISSE ERFORDERT.

Die Aufgabe:

Ein Grafikwürfel soll mit der Maus um seinen räumlichen Mittelpunkt drehbar sein, und zwar nach Möglichkeit in Echtzeit. Der Benutzer soll den Würfel so drehen können, wie wenn er mit der Hand an der ihm zugewandten Seite des Würfels entlangstriche. Die Drehachsen sind also nicht mit dem Würfel verbunden, sondern quasi mit dem Betrachter. Dies vergrößert den Rechenaufwand nicht unerheblich!

Die Implementation:

Eine Schleife (Abbruchbedingung: Tastendruck für Programmende; siehe Listing 1, *mouse_ctrl()*) fragt immer wieder die Mauscursor-Koordinaten ab und berechnet, wenn die Maustaste gedrückt ist, die Differenzen zu den zuletzt erhaltenen Werten. Sobald die Maustaste losgelassen wird, wartet die do-while-Schleife darauf, daß sie wieder gedrückt wird. Das Drücken der Maustaste entspricht also dem Anfassen des Würfels. Ein bestimmter Bruchteil dieser Differenzen läßt sich mit guter Näherung als Winkel übernehmen, um den der Würfel um die vertikale (x-Änderung) und horizontale Achse (y-Änderung) gedreht wurde. Da die Kantenlänge des Würfels 200 Pixel, der Abstand einer Kante vom Drehpunkt also Wurzel(2) * 100 = 141 Pixel beträgt, entsprechen einem Bogenmaß von 2π (eine volle Würfelumdrehung) $2\pi r = 889$ Pixel. Jede Drehung um ein Bogenmaß von 1 entspricht also einer Bewegung um r = 141 Pixel. Der Wert *ANGLESCALING* ist ein empirisch gefundener Wert in dieser Größenordnung. Der Würfel wird in Vektorgrafik dargestellt. Die Vektoren zeigen vom Ursprung zu den Eckpunkten. An dieser Stelle lohnt es sich zu fragen, wieviele Eckpunkte man eigentlich berechnen muß, um die Lage des Würfels eindeutig zu beschreiben. Wenn man die vier Eckpunkte einer Seite berechnet, kann man die jeweils räumlich gegenüberliegenden Eckpunkte durch Spiegeln am Mittelpunkt erhalten. Von diesen vier Eckpunkten kann man durch einen Trick wieder zwei einsparen (s.u.). (Genaugenommen reichen sogar ein Punkt und eine Information über einen zweiten Punkt, aber dies würde den Rahmen der Schulmathematik sprengen.)

Ich habe in diesem Falle die Koordinaten zweier auf einer Seite diagonal gegenüberliegender Eckpunkte (die Punkte vorne links und hinten rechts der am Anfang unten liegenden Seite, A und C) voreingestellt. Diese werden anhand der die Funktion erreichenden Änderungswinkel manipuliert (*new_cube()*; siehe Listing 2). Der aufmerksame Leser stellt hier fest: Wenn man den Würfel lange genug dreht, müßten die Rundungsfehler schließlich so groß werden, daß man am Ende keinen Würfel mehr vor sich hat, sondern einen Spat! Dies ist tatsächlich so; allerdings ist dieser Fehler so gering, daß er überhaupt nicht ins Auge fällt. Der Koordinatenursprung befindet sich in der räumlichen Mitte des Würfels, also im Drehpunkt. Zur Beschreibung der Lage der beiden Ur-Vektoren (MA und MC, wobei M der Ursprung ist) sind Polarkoordinaten am vorteilhaftesten, da die Information über die Bewegung in Winkelmaßen übergeben wird. Die Länge der Vektoren ist klar: Da die Kantenlänge des Würfels 200 Pixel beträgt, ist die Entfernung Mitte-Eckpunkt gleich 100 * Wurzel(3) = 173 Pixel.

Da es sich um ein dreidimensionales Koordinatensystem handelt, sind zur eindeutigen Beschreibung eines Vektors (außer der Länge) zwei Winkel erforderlich: Der Winkel Alpha beschreibt den Winkel um die Hochachse in der horizontalen Ebene (von oben betrachtet gegen den Urzeigersinn wachsend), der Winkel Beta um die horizontale Achse in der vertikalen Ebene, die senkrecht zur Bildschirmebene liegt (von links betrachtet gegen den Urzeigersinn wachsend). Der Vektor, dessen Winkel alle gleich Null sind, zeigt vom Ursprung direkt zum Betrachter. Zur Vereinfachung der Berechnung existiert noch der Winkel Phi, der den Betrag des Winkels eines Vektors zur Hochachse beschreibt ($0 \leq \text{Phi} \leq \pi/2$). Alle Winkel werden in Radians angegeben. Alpha und Phi von

A und C (a_A, a_C, phi_A, phi_C) sind statische Variablen, die so initialisiert werden, daß ein Würfel aufgespannt wird (s.o.). Diese Winkel werden bei jedem Aufruf von *new_cube*() auf den neuesten Stand gebracht. Dann werden die Polarkoordinaten in karthesische Koordinaten umgewandelt. Dabei bildet ein Koordinatentriplett von z.B. xA, yA und zA den Vektor MA.

Schließlich werden aus diesen beiden Punkten, die die Lage des Würfels im Raum ja schon eindeutig bestimmen, die Koordinaten der fehlenden Punkte der Grundseite, B und D, über das vektorielle Produkt ermittelt. Dazu werden die Vektoren MA und MC vektoriell miteinander multipliziert und das Ergebnis, Vektor OB, zum Vektor MO (vom Ursprung zum Mittelpunkt der Strecke AC) dazuaddiert. Hieraus erhält man den Vektor MB. Derselbe Vektor von MB subtrahiert ergibt den Vektor MD (was gleichbedeutend ist mit den Koordinaten von Punkt D, da M ja im Koordinatenursprung liegt). Es fällt auf, daß im Programmtext (siehe Listing 2) die Berechnung der y-Koordinate fehlt. Diese ist für die Grafik nicht mehr notwendig, da es sich um eine isometrische Darstellung handelt, also dreidimensional ohne Perspektive (Tiefenwirkung). Wie oben erklärt, können die räumlich gegenüberliegenden Eckpunkte durch Spiegeln am Ursprung erhalten werden. Dann brauchen nur noch die Offsets zum Bildschirmmittelpunkt addiert zu werden, und schon hat man die Bildschirmkoordinaten! Anschließend werden die Linien über die Line-A-Funktionen gezeichnet, was tatsächlich um einiges schneller geht als über GEM. Dies bringt leider mit sich, daß das Programm wegen der absoluten Bildschirmkoordinaten nur mit dem monochromen Monitor läuft. Es dürfte jedoch kein Problem sein, das Programm so umzuschreiben, daß es auch in den niedrigeren Auflösungen läuft, da die neuen Koordinaten erst ganz am Schluß bei der Umrechnung in zweidimensionale Bildschirmkoordinaten relevant sind.

Damit das Bild des rotierenden Würfels ohne störendes Wackeln und ohne Zerhacktwerden durch den Bildschirm-Elektronenstrahl erscheint, wird zu Beginn des Programms ein zweiter Bildschirmspeicher reserviert. Das Programm zeichnet nun das neue Bild in dem physikalisch jeweils gerade inaktiven Bildschirm, wartet auf den Rücklauf des Elektronenstrahls und schaltet dann die Bildschirme um. Ich habe hier bewußt auf eine detaillierte Beschreibung der mathematischen Grundlagen und die Umsetzung in den von mir verwendeten Algorithmus verzichtet. Es geht mir nicht darum, dieses Programm im Detail für jeden nachvollziehbar zu erklären, sondern ich habe nur die Lösung einiger interessanter Teilprobleme aufgezeigt, um Euch dazu anzuregen, Eurer Phantasie einmal freien Lauf zu lassen und mathematische Probleme in Angriff zu nehmen!

Optische Verbesserungen

Es ist kein Problem, aus dieser isometrischen Darstellung eine perspektivische zu machen. Die weiter hinten liegenden Punkte müssen nur näher an der Achse liegen, entlang der man blickt. Die Berechnung gestaltet sich in diesem Fall sehr einfach, da diese Achse mit der y-Achse identisch ist.

Man muß nur die Beträge der x- und z-Koordinaten abhängig von den y-Koordinaten verkleinern. Deshalb müssen nun bis zum Schluß alle Berechnungen in allen drei Dimensionen durchgeführt werden. Dazu muß zunächst der Variablensatz von "shorts" für die dreidimensionalen Koordinaten sämtlicher Eckpunkte vollständig sein. Anstelle der vier Zeilen an einschlägiger Stelle ist dann Listing 3 einzufügen. Der neue Rechenteil ist im Listing 4 zu finden. Die Konstante *PER* gibt die Entfernung des Punktes vom Ursprung an, in dem sich optisch alle zur y-Achse parallelen Geraden treffen. Sie muß natürlich als Konstante oben im Quelltext definiert werden. Folgender Wert ergibt einen akzeptablen perspektivischen Eindruck:

```
#define PER 1732
        /* entspricht 10 Radien */
```

Dank der geometrischen Besonderheiten der Würfelform ist es bei isometrischer Darstellung auch kein Problem, den Würfel ausgefüllt erscheinen zu lassen. Jede Hidden-Line-Ermittlung erübrigt sich: Es werden einfach alle Linien weggelassen, die zu dem/den hintersten Punkt/en führen. Das herauszubekommen, ist nicht weiter schwer. Dafür müssen natürlich wieder alle y-Koordinaten der räumlichen Lage mitberechnet werden. Der entsprechend einzusetzende Programmtext inklusive des neuen Zeichen-Blocks steht im Listing 5. Komplizierter wäre es allerdings, einen perspektivischen, ausgefüllten Würfel zu zeichnen - hier müßte man tatsächlich genau bestimmen, welche Kanten zu sehen sind und welche nicht.

Eine kleine Spielerei am Schluß: Ich finde, der Würfel sieht auf dem invertierten Bildschirm besser aus. Ich habe also ein kleines, universell verwendbares Programm geschrieben, das den Bildschirm invertiert. Es tut nichts anderes, als das niederwertigste Bit des Farbpalettenregisters 0 zu invertieren. Dies geht allerdings nur im Supervisormodus (Listing 6). Alle Programme wurden mit LATTICE C V3.04 geschrieben.



```
  1:  /* CUBEMAIN.C */
  2:
  3:  /* Drehbarer Vektorgrafik-Würfel, Kontrollteil */
  4:  /* (c) MAXON Computer GmbH
  5:
  6:  #include <stdlib.h>
  7:  #include <gemlib.h>
  8:  #include <osbind.h>
  9:  #include <linea.h>
 10:
 11:
 12:  #define ANGLESCALING 127.0     /* bestimmt die
                                       Relation von
                                       Mausbewegung
 13:                                   zu Würfeldrehung
                                      */
 14:
 15:  short phys_handle, handle;
 16:  short work_in[12], work_out[57];
 17:  short dummy;
 18:
 19:  void main(), init(), ausgang(), mouse_ctrl();
 20:  extern void new_cube();
 21:
 22:  int screen[2];
 23:  char *scrmem;
 24:
 25:
 26:  void main()
 27:  {
 28:     init();
 29:     mouse_ctrl();
 30:     ausgang();
 31:  }
 32:
 33:
 34:  void mouse_ctrl()
 35:  {
 36:     short mbstate, mx, my, mx_n, my_n;   /* alte u.
                                          neue(_n) Maus-
 37:                                      koordinaten */
 38:     double a_chg, b_chg;            /* Differenz der
                                          Mauskoord.  */
 39:
 40:
 41:     mbstate = mx = my = mx_n = my_n = 0;
 42:
 43:
 44:     do {            /* wartet auf linke Maustaste */
 45:        vq_mouse(handle, &mbstate, &mx, &my);
 46:     } while (!(mbstate && 1));
 47:     vq_mouse(handle, &mbstate, &mx_n, &my_n);
                     /* holt aktuelle Mauskoord. */
 48:
 49:     while (!kbhit()) {
 50:        a_chg = (double)(mx_n - mx) / ANGLESCALING;
                     /* berechnet Winkelände-  */
 51:        b_chg = (double)(my - my_n) / ANGLESCALING;
                     /* rung in Radians        */
 52:        new_cube(&a_chg, &b_chg);
                     /* neuen Würfel zeichnen! */
 53:
 54:        mx = mx_n;
 56:        my = my_n;
 57:        vq_mouse(handle, &mbstate, &mx_n, &my_n);
 58:        if (mbstate && 1)  /* Mausknopf gedrückt? */
 59:           continue;       /* gleich weitermachen */
 60:
 61:        do {               /* ansonsten wieder    */
 62:           vq_mouse(handle, &mbstate, &mx, &my);
                              /* auf Mausknopf warten*/
 63:           if (kbhit()) break;     /* siehe oben. */
 64:        } while (!(mbstate && 1));
 65:        vq_mouse(handle, &mbstate, &mx_n, &my_n);
 66:     }
 67:
 68:     getch();
 69:  }
 70:
 71:
 72:  void init()
 73:  {
 74:     int i;
 75:     double dnull = 0.0;
 76:     short muster = 0xFFFF;
 77:
 78:     appl_init();          /* GEM-Initialisierung */
 79:     for (i = 0; i <= 9; work_in[i++] = 1);
 80:     work_in[10] = 2;
 81:
 82:     phys_handle = graf_handle(&dummy, &dummy,
                                    &dummy, &dummy);
 83:     handle = phys_handle;
 84:     v_opnvwk(work_in, &handle, work_out);
 85:     v_hide_c(handle);
 86:     v_clrwk(handle);
 87:
 88:     screen[0] = Physbase();    /* zweiten Bildschirm
                                       holen */
 89:     if ((scrmem = calloc(8064, 4)) == NULL) {
 90:        form_alert(1,
 91:          "[3][Nicht genug|Speicherplatz|
                  für den zweiten|Bildschirm!][Mist]");
 92:        ausgang();
 93:     }
 94:     screen[1] = (int)scrmem + 256;
 95:     screen[1] -= screen[1] % 256;
 96:
 97:     linea0();              /* LineA-Initialisierung */
 98:
 99:     LSTLIN = -1;
100:     LNMASK = 0xFFFF;
101:     WMODE  = 0;
102:     PATPTR = &muster;
103:     PATMSK = 0;
104:     CLIP   = 0;
105:
106:     new_cube(&dnull, &dnull);     /* Ausgangswürfel
                                         zeichnen */
107:
108:     graf_mouse(FLAT_HAND, 0);
109:     v_show_c(handle, 1);
110:  }
111:
112:
113:  void ausgang()
114:  {
115:     Setscreen(screen[0], screen[0], -1);
    /* wieder auf Originalbildschirm zurückschalten */
117:     free(scrmem);
118:
119:     v_show_c(handle, 0);
120:     v_clsvwk(handle);
121:     appl_exit();
122:
123:     exit(0);
124:  }
```

Listing 1

```
  1:  /* CUBE.C */
  2:
  3:  /* Drehbarer Vektorgrafik-Würfel, Rechen- und
        Zeichenteil */
  4:  /* (c) MAXON Computer GmbH
  5:
  6:  #include <math.h>
  7:  #include <linea.h>
  8:  #include <osbind.h>
  9:
 10:  #define RADIUS 173.20508075  /* Entfernung vom
                     Ursprung zum Eckpukt (Pixel) */
 11:  #define RAD_Q 30000.0 /* Quadrat des Radius    */
 12:  #define XM 320        /* Lage des Ursprungs in
                              zweidimensionalen     */
 13:  #define YM 200        /* Bildschirmkoordinaten */
 14:  #define sign(x) (((x)<0)?(-1):(1))
 15:  #define PREC 4.0
 16: /* empirisch ermittelter Wert, der angibt, in
       welchem Bereich angenommen wird, daP
 17:   sowohl Alpha als auch Beta von A oder C
 18:   gleich Null sind, so daP eine Ausnahme-
 19:   regelung getroffen werden kann.
 20:   Ansonsten würde der Würfel in diesem Fall
 21:   in der Grafik buchstäblich zusammenklappen!  */
 22:
 23:  void new_cube(), clear();
 24:
 25:  extern int screen[2];
 26:
 27:
 28:  void new_cube(a_chg, b_chg)
```

```
29:  double *a_chg, *b_chg;
30:  {
31:     static double a_A = -PI/4.0;
            /* Deklaration der Variablen  */
32:     double b_A;   /* sowie ggf. Initialisierung */
33:     static double a_C = 3.0*PI/4.0;
34:     double b_C;
            /* Systematik der Variablen-  */
35:     static double phi_A = -0.61547970865;
            /* namen:                     */
36:     static double phi_C = -0.61547970865;
            /* Großbuchstaben sind Eck-   */
37:     double x_A, x_C;
            /* te; O ist der Mittelpunkt  */
38:  /*---------Anfang Listing 3--------*/
39:     short  xO,    zO, xOB,    zOB;
            /* der Strecke AC.            */
40:     short  xA, yA, zA, xB,     zB;
            /* a, b und phi sind die Win- */
41:     short  xC, yC, zC, xD,     zD;
            /* kel Alpha, Beta und Phi.   */
42:     short  xE, zE, xF, zF, xG, zG, xH, zH;
43:  /*---------Ende Listing 3----------*/
44:     char help;
45:     double cos_b_A;
46:     double tan_b_A;
47:     double cos_b_C;
48:     double tan_b_C;
49:     short abs_zA, abs_zC;
50:     double abs_x_A, abs_x_C;
51:
52:
53:     a_A += *a_chg;          /* A - Winkel update */
54:
55:     b_A = atan2(tan(phi_A), cos(a_A));
56:     if ((help=sign(phi_A)) != sign(b_A))  b_A +=
                                  PI * help;
57:
58:     x_A = RADIUS * cos(phi_A) * sin(a_A);
59:
60:     b_A += *b_chg;
61:
62:     a_A = atan2(x_A, sqrt(RAD_Q-x_A*x_A) *
              abs(cos_b_A=cos(b_A)));
63:     if (cos_b_A < 0)  a_A = sign(x_A) * PI - a_A;
64:
65:     phi_A = atan((tan_b_A=tan(b_A)) * cos(a_A));
66:
67:
68:     a_C += *a_chg;          /* C - Winkel update */
69:
70:     b_C = atan2(tan(phi_C), cos(a_C));
71:     if ((help=sign(phi_C)) != sign(b_C))  b_C +=
                                  PI * help;
72:
73:     x_C = RADIUS * cos(phi_C) * sin(a_C);
74:
75:     b_C += *b_chg;
76:
77:     a_C = atan2(x_C, sqrt(RAD_Q-x_C*x_C) *
              abs(cos_b_C=cos(b_C)));
78:     if (cos_b_C < 0)  a_C = sign(x_C) * PI - a_C;
79:
80:     phi_C = atan((tan_b_C=tan(b_C)) * cos(a_C));
81:
82:
83:     xA = (short)x_A;
84:     xC = (short)x_C;
85:     zA = (short) - RADIUS * sin(phi_A);
                        /* restliche Koordinaten */
86:     yA = (short) ((abs_zA=abs(zA)) >
         (abs_x_A=abs(x_A)))
87:          ? ((abs_zA  > PREC) ? (zA / tan_b_A) : (-
               sign(cos_b_A)*RADIUS))
88:          : ((abs_x_A > PREC) ? (- x_A / tan(a_A))
             : (-sign(cos_b_A)*RADIUS));
89:     zC = (short) - RADIUS * sin(phi_C);
90:     yC = (short) ((abs_zC=abs(zC)) >
        (abs_x_C=abs(x_C)))
91:          ? ((abs_zC  > PREC) ? (zC / tan_b_C) : (-
               sign(cos_b_C)*RADIUS))
92:          : ((abs_x_C > PREC) ? (- x_C / tan(a_C))
             : (-sign(cos_b_C)*RADIUS));
93:
94:  /*--------Anfang Listing 4 und 5---------------*/
95:
96:     xO = (xA+xC) / 2;                /* Vektor MO   */
97:     zO = (zA+zC) / 2;
98:
99:     xOB = (yA*zC - zA*yC) / 200;  /* Vektor OB   */
100:    zOB = (xA*yC - yA*xC) / 200;
101:
102:    xB = xO + xOB;                 /* Koord von B*/
103:    zB = zO + zOB;
104:
105:    xD = xO - xOB;                 /* Koord von D*/
106:    zD = zO - zOB;
107:
108:    xE = -xC + XM;  zE = -zC + YM;
                               /* Koords spiegeln   */
109:    xF = -xD + XM;  zF = -zD + YM;
110:    xG = -xA + XM;  zG = -zA + YM;
111:    xH = -xB + XM;  zH = -zB + YM;
112:
113:    xA += XM;  zA += YM;    /* fehlende Offsets */
114:    xB += XM;  zB += YM;
115:    xC += XM;  zC += YM;
116:    xD += XM;  zD += YM;
117:
118: /*-----------Ende Listing 4-------------*/
119:
120:
121:    {                           /* zeichnen        */
122:       static int scrno = 0;
123:
124:       scrno ^= 1;
125:       lineaa();             /* Mauscursor weg */
126:       Setscreen(screen[scrno], -1, -1);
                    /* logischen Bildschirm wechseln */
127:       clear();
128:       COLBIT0 = 1;
129:       LNMASK = 0xFFFF;
130:
131:       X1 = xA;  Y1 = zA;
132:       X2 = xB;  Y2 = zB;
133:       linea3();
134:
135:       X2 = xD;  Y2 = zD;
136:       linea3();
137:
138:       X2 = xE;  Y2 = zE;
139:       linea3();
140:
141:       X1 = xC;  Y1 = zC;
142:       X2 = xB;  Y2 = zB;
143:       linea3();
144:
145:       X2 = xD;  Y2 = zD;
146:       linea3();
147:
148:       X2 = xG;  Y2 = zG;
149:       linea3();
150:
151:       X1 = xF;  Y1 = zF;
152:       X2 = xB;  Y2 = zB;
153:       linea3();
154:
155:       X2 = xE;  Y2 = zE;
156:       linea3();
157:
158:       X2 = xG;  Y2 = zG;
159:       linea3();
160:
161:       X1 = xH;  Y1 = zH;
162:       X2 = xD;  Y2 = zD;
163:       linea3();
164:
165:       X2 = xE;  Y2 = zE;
166:       linea3();
167:
168:       X2 = xG;  Y2 = zG;
169:       linea3();
170:
171: /*----------Ende Listing 5--------------*/
172:
173:       Vsync();
174:       Setscreen(-1, screen[scrno], -1);
                     /* physikalischer Bildschirm */
175:       linea9();   /* Mauscursor zeigen */
176:    }
177: }
178:
179: void clear()       /* löscht genau rechteckigen */
180: {                  /* Bildschirmber., dv Würfel */
```

```
181:     COLBIT0 = 0;   /* maximal genommen wird     */
182:
183:     X1 = (short)XM-RADIUS-1;
184:     Y1 = (short)YM-RADIUS-1;
185:     X2 = (short)XM+RADIUS+1;
186:     Y2 = (short)YM+RADIUS+1;
187:
188:     linea5();
189: }
```

Listing 2

```
 1:  (1. Änderung perspektivische
      Darstellung):
 2:
 3:
 4:     short  xO, yO, zO, xOB, yOB, zOB;
 5:     short  xA, yA, zA, xB,  yB,  zB;
 6:     short  xC, yC, zC, xD,  yD,  zD;
 7:     short  xE, yE, zE, xF, yF, zF, xG, yG, zG, xH,
                yH, zH;
 8:
 9:
10:
11:
12:  (2. Änderung perspektivische
                 Darstellung):
13:
14:
15:     xO = (xA+xC) / 2;              /* Vektor MO */
16:     yO = (yA+yC) / 2;
17:     zO = (zA+zC) / 2;
18:
19:     xOB = (yA*zC - zA*yC) / 200;  /* Vektor OB */
20:     yOB = (zA*xC - xA*zC) / 200;  /* über vekto-
                                   rielles Produkt */
21:     zOB = (xA*yC - yA*xC) / 200;
22:
23:     xB = xO + xOB;              /* Koords von B */
24:     yB = yO + yOB;
25:     zB = zO + zOB;
26:
27:     xD = xO - xOB;              /* Koords von D */
28:     yD = yO - yOB;
29:     zD = zO - zOB;
30:
31:     xE = -xC;  yE = PER-yC;  zE = -zC;  /* Koords
                                          spiegeln */
32:     xF = -xD;  yF = PER-yD;  zF = -zD;
33:     xG = -xA;  yG = PER-yA;  zG = -zA;
34:     xH = -xB;  yH = PER-yB;  zH = -zB;
35:
36:     yA += PER; yB += PER; yC += PER; yD += PER;
37:
38:     xA *= (PER-RADIUS) / yA;  zA *= (PER-RADIUS) /
               yA;
39:     xB *= (PER-RADIUS) / yB;  zB *= (PER-RADIUS) /
               yB;
40:     xC *= (PER-RADIUS) / yC;  zC *= (PER-RADIUS) /
               yC;
41:     xD *= (PER-RADIUS) / yD;  zD *= (PER-RADIUS) /
               yD;
42:     xE *= (PER-RADIUS) / yE;  zE *= (PER-RADIUS) /
               yE;
43:     xF *= (PER-RADIUS) / yF;  zF *= (PER-RADIUS) /
               yF;
44:     xG *= (PER-RADIUS) / yG;  zG *= (PER-RADIUS) /
               yG;
45:     xH *= (PER-RADIUS) / yH;  zH *= (PER-RADIUS) /
               yH;
46:
47:     xA += XM;  zA += YM;    /* fehlende Offsets */
48:     xB += XM;  zB += YM;
49:     xC += XM;  zC += YM;
50:     xD += XM;  zD += YM;
51:     xE += XM;  zE += YM;
52:     xF += XM;  zF += YM;
53:     xG += XM;  zG += YM;
54:     xH += XM;  zH += YM;
```

Listing 3 u. 4

```
 1:  (Änderung ausgefüllter Würfel):
 2:
 3:
 4:     xO = (xA+xC) / 2;               /* Vektor MO */
 5:     yO = (yA+yC) / 2;
 6:     zO = (zA+zC) / 2;
 7:
 8:     xOB = (yA*zC - zA*yC) / 200;    /* Vektor OB */
 9:     yOB = (zA*xC - xA*zC) / 200;
10:     zOB = (xA*yC - yA*xC) / 200;
11:
12:     xB = xO + xOB;                /* Koords von B */
13:     yB = yO + yOB;
14:     zB = zO + zOB;
15:
16:     xD = xO - xOB;                /* Koords von D */
17:     yD = yO - yOB;
18:     zD = zO - zOB;
19:
20:     xE = -xC + XM;  yE = -yC;  zE = -zC + YM;
                                /* Koords spiegeln */
21:     xF = -xD + XM;  yF = -yD;  zF = -zD + YM;
22:     xG = -xA + XM;  yG = -yA;  zG = -zA + YM;
23:     xH = -xB + XM;  yH = -yB;  zH = -zB + YM;
24:
25:     xA += XM;  zA += YM;     /* fehlende Offsets */
26:     xB += XM;  zB += YM;
27:     xC += XM;  zC += YM;
28:     xD += XM;  zD += YM;
29:
30:     ymax = yA;           /* Finden des gröPten y */
31:     if (yB > ymax) ymax = yB;
32:     if (yC > ymax) ymax = yC;
33:     if (yD > ymax) ymax = yD;
34:     if (yE > ymax) ymax = yE;
35:     if (yF > ymax) ymax = yF;
36:     if (yG > ymax) ymax = yG;
37:     if (yH > ymax) ymax = yH;
38:
39:
40:
41:     {                               /* zeichnen */
42:       static int scrno = 0;
43:
44:       scrno ^= 1;
45:       lineaa();
46:       Setscreen(screen[scrno], -1, -1);
47:       clear();
48:       COLBIT0 = 1;
49:       LNMASK = 0xFFFF;
50:
51:       if (ymax != yA) {
52:          X1 = xA;  Y1 = zA;
53:
54:          if (ymax != yB) {
55:             X2 = xB;  Y2 = zB;
56:             linea3();
57:          }
58:
59:          if (ymax != yD) {
60:             X2 = xD;  Y2 = zD;
61:             linea3();
62:          }
63:
64:          if (ymax != yE) {
65:             X2 = xE;  Y2 = zE;
66:             linea3();
67:          }
68:       }
69:
70:       if (ymax != yC) {
71:          X1 = xC;  Y1 = zC;
72:
73:          if (ymax != yB) {
74:             X2 = xB;  Y2 = zB;
75:             linea3();
76:          }
77:
78:          if (ymax != yD) {
79:             X2 = xD;  Y2 = zD;
80:             linea3();
81:          }
82:
83:          if (ymax != yG) {
84:             X2 = xG;  Y2 = zG;
85:             linea3();
86:          }
```

```
 87:         }
 88:
 89:         if (ymax != yF) {
 90:            X1 = xF;   Y1 = zF;
 91:
 92:            if (ymax != yB) {
 93:               X2 = xB;   Y2 = zB;
 94:               linea3();
 95:            }
 96:
 97:            if (ymax != yE) {
 98:               X2 = xE;   Y2 = zE;
 99:               linea3();
100:            }
101:
102:            if (ymax != yG) {
103:               X2 = xG;   Y2 = zG;
104:               linea3();
105:            }
106:         }
107:
108:         if (ymax != yH) {
109:            X1 = xH;   Y1 = zH;
110:
111:            if (ymax != yD) {
112:               X2 = xD;   Y2 = zD;
113:               linea3();
114:            }
115:
116:            if (ymax != yE) {
117:               X2 = xE;   Y2 = zE;
118:               linea3();
119:            }
120:
121:            if (ymax != yG) {
122:               X2 = xG;   Y2 = zG;
123:               linea3();
124:            }
125:         }
```

Listing 5

Eingabe mit Stil

M. G. Berberich

Wer schon einmal probiert hat, eine Formulareingabe in GFA-Basic zu schreiben, kennt das Problem: Die INPUT-, LINE INPUT- und FORM INPUT-Befehle des GFA-BASIC lassen zwar eine komfortable Eingabe zu, aber man kann sie nur durch RETURN beenden. Wenn man die Eingabe nun zum Beispiel mit der "Cursor hoch" - oder "Cursor runter"-Taste verlassen will, um ins nächste bzw. vorige Feld zu gelangen, gibt es nur eine Möglichkeit: eine eigene Eingaberoutine muss her.

Dabei ergibt sich ein Problem: Wenn man einen String normal an eine Prozedur übergibt, kann man ihn nicht zurückübergeben. Wenn man ihn aber mit **String* übergibt, kann man ihn zurückgeben, aber nicht verwenden. Also muß eine Methode her um einen String hin- und rückzuübergeben. Hier die Lösung des 'Hin-rück-Problems'. Von dem String liegt nur die Adresse vor. Um also an ihn zu gelangen, muß man ihn zuerst in einen localen String in ausreichender Größe kopieren. Dies erledigt Listing 1. Wie funktioniert die Sache? Die Prozedur erhält die Adresse des Arrptr des Strings. Zuerst werden die nötigen Hilfsvariablen als Local definiert. Als nächstes wird die Länge des 'übergebenen' Strings bestimmt. Sie steht in *Arrptr+4*. Dann werden der Hilfs-String auf die nötige Länge mit Spaces aufgefüllt und der *Varrptr* des Strings und des Hilfs-Strings bestimmt. Beim Hilfs-String geht das mit *VARRPTR*, beim Übergabestring muß man die Adresse aus *Arrptr* holen. Nun kann man den Inhalt des Strings byteweise kopieren. Als Resultat hat man den String im Hilfs-String und kann ihn verarbeiten. Auf diese Art und Weise kann man einen String in eine Prozedur hin- und rückübergeben.

Als Beispiel habe ich die oben angesprochene Input-Routine verwendet (Listing 2). Sie arbeitet im Einfügemodus, was bedeutet, daß der Buchstabe, der gerade eingegeben wird, keinen anderen überschreibt, sondern an der Cursor-Position hineingeflickt wird. Die Taste Backspace löscht den links neben dem Cursor liegenden Buchstaben, die Taste Delete den rechts neben dem Cursur. Die Taste Esc löscht das gesamte Eingabefeld. Mit den Tasten "Cursor hoch", "Cursor runter" und "Return" bzw "Enter" wird die Eingabe verlassen. Der Code der Taste, die den Abbruch verursacht hat, wird zusammen mit dem String zurückgegeben.

Der Syntax der Prozedur lautet *@input(v$,*i$,l,C,*r)*. Der erste Parameter *v$* ist ein Prompttext. Er wird auf den Bildschirm geschrieben und kann nicht geändert werden. Der zweite Parameter *I$* ist der Eingabe-String. Sein Inhalt wird ebenfalls angezeigt (Trick von oben), kann aber geändert werden. Nach dem procedure-Aufruf enthält *i$* den geänderten Text. Es muß **i$* heißen, da nicht der Wert von *I$*, sondern die Adresse übergeben wird. *L* ist die Länge des Strings. Der Rückgabe-String wird mit Space genau auf diese Länge ergänzt. *C* gibt die Position an, die der Cursor am Anfang der Eingabe haben soll. In *R* wird der Code der Taste zurückgeliefert, die zum Abbruch geführt hat.

In der Exit-Zeile können selbstverständlich noch beliebige andere Codes eingegeben werden, die zum Abbruch führen sollen. Es ist der Code zu verwenden, den *INP(2)* liefert.



```
 1:  '          Stringübergabe für GFA-Basic
 2:  '          by M.G.Berberich
 3:  '          (c) 1990 MAXON Computer GmbH
 4:  '
 5:  INPUT "Name ",a$
 6:  @gruss(*a$)                         ! proc Aufruf
 7:  PRINT a$
 8:  PROCEDURE gruss(p.string%)
 9:    LOCAL memo$,l%,ptrm%,ptrs%,a% ! Arbeitsvars
10:    l%=DPEEK(p.string%+4)         ! String-Länge
11:    memo$=STRING$(l%," ")         ! Leerstring
                                       erzeugen
12:    ptrm%=VARPTR(memo$)           ! memo$-Adresse
13:    ptrs%=LPEEK(p.string%)        ! p.-st.-Adr.
14:    FOR a%=1 TO l%                ! \
15:      POKE ptrm%,PEEK(ptrs%)      !  \
16:      INC ptrm%                   !   Kopieren
17:      INC ptrs%                   !  /
18:    NEXT a%                       ! /
19:    *p.string%="Hallo "+memo$
20:  RETURN
```

```
 1:  '         Inputfunktion für GFA-Basic
 2:  '         by M.G.Berberich
 3:  '         (c) 1990 MAXON Computer GmbH
 4:  '
 5:  ' Syntax:
 6:  ' @input(v$,*i$,l,C,*r)
 7:  ' v$ = prompttext
 8:  ' i$ = Vorgabe Text
 9:  ' l  = Länge des Strings
10:  ' c  = Cursorposition relativ im String
11:  ' r  = Taste die zum Abbruch
12:  '      geführt hat
13:  '
14:  PRINT
15:  PRINT
16:  PRINT
17:  in$="Na sowas"
18:  @input("Eingabe: ",*in$,20,5,*key)
19:  PRINT
20:  PRINT in$
21:  '
22:  PROCEDURE input(t$,i%,l%,cp%,rw%)
23:    LOCAL a%,b%,c%,d%,i$
24:    i$=STRING$(l%," ")          !
25:    b%=DPEEK(i%+4)              !
26:    b%=MIN(b%,l%)               !Abgewandelte
27:    s%=VARPTR(i$)               !  Version
28:    c%=s%                       !der Methode
29:    d%=LPEEK(i%)                !     von
30:    FOR a%=1 TO b%              ! Listing 2
31:      POKE c%,PEEK(d%)          !
32:      INC c%                    !
33:      INC d%                    !
34:    NEXT a%                     !
35:    PRINT t$;
36:    c%=CRSLIN
37:    b%=CRSCOL
38:    PRINT CHR$(27);"e";i$;      ! Cursor an + i$
39:    PRINT AT(b%+cp%,c%);
40:    DEC l%
41:    DO
42:      a%=INP(2)
43:      EXIT IF a%=13 OR a%=200 OR a%=208
44:      IF a%=8 AND cp%>0  ! Backspace
45:        PRINT CHR$(27);"D";RIGHT$(i$,l%-cp%+1);" ";
46:        DEC cp%
47:        PRINT AT(b%+cp%,c%);
48:        i$=LEFT$(i$,cp%)+RIGHT$(i$,l%-cp%)+" "
49:      ENDIF
50:      IF a%=127        ! Delete
51:        PRINT RIGHT$(i$,l%-cp%);" ";
52:        PRINT AT(b%+cp%,c%);
53:        i$=LEFT$(i$,cp%)+RIGHT$(i$,l%-cp%)+" "
54:      ENDIF
55:      IF a%=27         ! Esc = Lösche alles
56:        i$=STRING$(l%," ")
57:        cp%=0
58:        PRINT AT(b%,c%);i$
59:        PRINT AT(b%,c%);
60:      ENDIF
61:      IF a%=205 AND cp%<l% ! Cursor rechts
62:        INC cp%
63:        PRINT CHR$(27);"C";
64:      ENDIF
65:      IF a%=203 AND cp%>0  ! Cursor links
66:        DEC cp%
67:        PRINT CHR$(27);"D";
68:      ENDIF
69:      IF cp%<=l% AND a%>31 AND a%<127  !Eingabe-
                                            zeichen
70:        INC cp%
71:        PRINT CHR$(a%);MID$(i$,cp%,l%-cp%+1);
72:        PRINT AT(b%+cp%,c%);
73:        i$=LEFT$(i$,cp%-1)+CHR$(a%)+MID$(i$,cp%,
                                           l%-cp%+1)
74:      ENDIF
75:    LOOP
76:    *i%=i$
77:    *rw%=a%
78:    PRINT CHR$(27);"f";   ! Cursor aus
79:  RETURN
```

DIE HINTERGRUND-PROGRAMMIERUNG VON GRAFIK

oder: Wie arbeiten zwei verschiedene Spezialisten miteinander an einem guten Projekt.

Gregor Schabrun

WIR BEFINDEN UNS IM JAHR 1503. SEIT DIESER GUTENBERG DIE DRUCKKUNST ERFUNDEN HAT, EXISTIERT DAS ALLGEMEINE BEDÜRFNIS, MASSLOS ZEITGENÖSSISCHE EREIGNISSE IN BÜCHERN FESTZUHALTEN. DAS INTERESSE AN EINEM EIGENEN DRUCKER IN JEDEM GEHOBENEN HAUSE HAT SOLCH GROSSE DIMENSIONEN ANGENOMMEN, DASS SICH UNSERE ZWEI FREUNDE MANFRED VON DER CLEVERBURG UND WILHELM VON FAULENZFURCHT VOR KURZEM INS DRUCKERGESCHÄFT GESTÜRZT HABEN.

Cleverburg ist ein richtiger Manager und Verkäufer des Mittelalters, während von Faulenzfurcht ein wahrer Meister der Kunst des Druckerherstellens ist. Die Nachfrage ist groß, so daß sie jeden fertiggestellten Drucker verkaufen können. Ihre Arbeitsweise ist folgendermaßen zu beschreiben: Sie arbeiten an zwei verschiedenen Tischen. Von Faulenzfurcht baut einen Drucker, den er dann recht mühsam auf den Verkaufstisch schiebt, was durchschnittlich etwa eine halbe Stunde in Anspruch nimmt. Ist der Drucker einmal auf Cleverburgs Tisch, kann er von diesem verkauft werden. Ein paar Urkunden erstellt und unterzeichnet - und fertig ist die Sache! Unterdessen hat von Faulenzfurcht schon mit dem nächsten Drucker begonnen. Irgendwie hat man den Eindruck, daß von Faulenzfurcht etwa viermal soviel arbeitet wie Cleverburg, der ja die meiste Zeit wartet. Bei sechs Druckern am Tag braucht die Herumschieberei der fertigen Drucker auf den anderen Tisch ein Drittel des Arbeitstages.

Angesichts dessen und der Tatsache, daß die Nachfrage nach Druckern dauernd wächst, entschließen sich die beiden eines Tages, ihre Arbeitsteilung zu überarbeiten. Cleverburgs geniale Intuition löst dann prompt auch das Problem. Seine Lösung: Statt sie den Drucker von einem Tisch auf den anderen zu schieben, begibt sich Cleverburg immer an den Tisch, an dem der Drucker gerade fertig geworden ist. Da dabei nur ein paar Urkunden den Platz wechseln müssen und nicht wie vorher der dicke Drucker, scheint das die einzig richtige Lösung zu sein. Der Arbeitsvorgang sieht dann so aus:
Von Faulenzfurcht baut den Drucker am Tisch A. Cleverburg verkauft den letzten Drucker oder hat ihn schon verkauft am Tisch B. Sobald der Drucker fertig ist (am Tisch A), zieht Cleverburg zu diesem Tisch und verkauft ihn. Kurz darauf beginnt von Faulenzfurcht, einen neuen Drukker zu bauen am Tisch B. Übersichtshalber vertauschen wir jetzt die Bezeichnungen der Tische A und B, und der Arbeitszyklus kann wieder von vorn beginnen. Das Resultat dieser neuen Arbeitsmethode: die halbe Stunde mühsame Herumschieberei pro Drucker entfällt völlig. Außerdem ist diese Methode auch verallgemeinerungsfähig. Es spielt hier keine Rolle, ob der Drukker groß ist oder ob es mehrere Drucker sind. Es ist immer leichterm, Urkunden herumzuschieben als Drucker.

Im ATARI ST und auch in jedem anderen Computer spielt sich die gleiche Geschichte ab. Unsere zwei Freunde Cleverburg und von Faulenzfurcht sind dabei nichts anderes als der Video-Chip und der gute alte 68000er! Die Drucker entsprechen den aufwendigen Grafiken, Bildern einer Bildsequenz, eines Films, eben einer bewegten Grafik. Die beiden Arbeitstische A und B, das sind zwei Bildschirme, die der Videoprozessor wahlweise darstellen kann. Der Tisch, an dem sich Cleverburg befindet, ist der Bildschirm, der vom

Monitor dargestellt wird, der Tisch an dem von Faulenzfurcht arbeitet, ist der Bildschirm, den man nicht sieht, aber an dem das nächste Bild erstellt wird.

Trockentheorie

Der ATARI ST unterscheidet zwischen logischem und physikalischem Bildschirm. Gezeichnet wird grundsätzlich im logischen Bildschirm. Auf dem Monitor dargestellt wird aber der Speicherbereich ab physikalischer Bildschirmadresse. Nach dem Einschalten des Rechners decken sich die beiden Bildschirme, d.h. die Startadresse des physikalischen Bildschirms ist gleich der des logischen. Ermittelt wird die logische Bildschirmadresse mit der XBIOS(3) Funktion. Die physikalische Bildschirmadresse bekommt man mit XBIOS(2). Wenn Sie das ausprobieren, erhalten Sie zwei identische Zahlen (Bildschirmadressen). Das heißt, jeder ausgeführte Grafikbefehl zeigt seine Wirkung direkt auf dem Monitor (z.B. BOX, CIRCLE etc.). Das muß aber nicht immer so sein. Man kann eine oder beide Bildschirmadressen ändern, beispielsweise so, daß sich die beiden Bildschirme nicht decken und nicht überlappen. Das geschieht mit der Funktion XBIOS(5).

8ung!

Bevor Sie jetzt aber mit dem Herumpröbeln beginnen, möchte ich Sie noch warnen: Der Videoprozessor ist ein bißchen wählerisch und frißt nicht jede Adresse. Im Klartext heißt das: Die Adresse, an der der Videoprozessor zugreifen soll, muß durch 256 teilbar sein. Hier spricht meine Erfahrung, die ich gemacht habe, als ich genau dies nicht beachtete. Es gab grausame Abstürze, an denen sogar OMIKRONs Reset-Festigkeit nichts ändern konnte.

Anwendung der Trockentheorie

Die mit der Methode der Hintergrundprogrammierung von Grafik gewonnene Erkenntnis implementiert viele Tricks, von denen ich einen erläutern möchte. Da wäre also die SWITCH-Technik, von der unsere zwei Freunde Cleverburg und von Faulenzfurcht zu berichten wissen. Das Kochrezept sieht so aus:

1. Man sorgt für 32k freien Speicher, der nicht von außen gestört werden darf, und an einer durch 256 teilbaren Adresse anfängt. Die Adresse dieses Bildschirms nennen wir *Ad1*.
2. Durch XBIOS(2) erhält man die Adresse der 32k des üblichen Bildschirms. Sie wird in *Ad2* gespeichert.
3. Man vertauscht die beiden Variablen *Ad1* und *Ad2* (z.B. mit SWAP).
4. Man ruft XBIOS(5) auf mit Ad1 für logische und *Ad2* für physikalische Adresse.
5. Man erstellt eine Grafik mit den normalen Befehlen (CIRCLE, BOX etc.).
6. Man fährt weiter bei Nr.3.

Dabei ist zu beachten:

- Das CLIPPING RECTANGLE muß eingestellt werden!

- Bei Nr.5 ist evtl. ein CLS schneller als das Rücksetzen aller unerwünschten alten Grafikelemente.

- Logisch bedingt blitzt der Bildschirm ab und zu auf, was auf eine Art "Eigenresonanz" des Videoprozessors zurückzuführen ist. Dies kann man verhindern, indem man zyklisch nach einer gewissen Zeitspanne in der Größenordnung halbe Minute einen *Wait For Vertikal Blank* einpflanzt (OMIKRON: WVBL, GFA:VSYNC).

STARRIDE oder das BEISPIELPROGRAMM arbeiten unter genau der eben vorgestellten SWITCH-Technik. Die dazu wichtigen Zeilen werden wie folgt erklärt:

Zeile 10 ermittelt die physikalische Bildschirmadresse und übergibt sie der Variablen *Ad1* (wie unter 1. in unserem Kochrezept). Zeile 11 reserviert 32k Speicher im GEMDOS-Speicher, der vorher mit CLEAR 100000 voreingestellt wurde, und benutzt die durch *MEMORY* gewonnene Adresse um die am nächsten liegende durch 256 teilbare Adresse zu errechnen. Die errechnete, korrekte Adresse wird in *Ad2* gespeichert (Kochrezept: Nr.2). Übergibt man vorher einer Variablen *Dumm$* den String *SPACE$(32256)*, und ersetzt man *MEMORY(32256)* durch *VARPTR(Dumm$)*, so kann man die Zeile 11 auch für GFA benutzen. Die Prozedur *Switch_Bildschirm* führt den Bildschirmswitch durch, danach sind logische und physikalische Bildschirmadresse vertauscht (entspricht Punkt 3 und 4 unseres Kochrezepts). Nach jedem alten Bild muß vor dem Erstellen der neuen Grafik ein CLS erfolgen. Anstelle des CLS wird aber ein den ganzen Bildschirm füllenden Background, der in der Prozedur *Kreiere_Hintergrund* erstellt und im Speicher ab Adresse Back gesichert wurde, hineingesetzt (OMIKRON: MEMORY_-MOVEB; GfA: BMOVE oder evtl. auch SGET und SPUT).

Zeile 5 killt den immerblinkenden Cursor. Bemerkenswert auch für GFA-Programmierer ist die Zeile 4.

Zur Grafik selbst: Die bewegte Grafik, die ich im BEISPIELPROGRAMM gewählt habe, besteht aus ganz vielen Sternchen, die auf den Zuschauer zukommen, bzw. das erweckt den Eindruck, als ob man in einem Raumschiff durch Sterngalaxien gleiten würde. Das wird häufig in Intros usw. benutzt. Didaktisch ist das vielleicht eher ein kompliziertes Beispiel, aber da es aus recht übersichtlichen Prozeduren besteht, sollte der Gedankengang nachvollziehbar sein.

Die Kinematik eines Sterns besteht aus einer Koordinate und einer momentanen Geschwindigkeit. Die Bewegung des Sterns beginnt ungefähr in der Mitte des Bildschirms mit einer zugeteilten Anfangsgeschwindigkeit. Und das alles in x- wie auch in y- Richtung. Je mehr sich der Stern vom Mittelpunkt entfernt, umso mehr wird er nach außen beschleunigt. D.h. die Beschleunigung ist abhängig von der Entfernung. Das ergibt eine Matrix, die ganz am Anfang des Programms alle möglichen Beschleunigungen in Abhängigkeit aller möglichen Entfernungen (auf dem Bildschirm ist ja die Anzahl aller möglicher Entfernungen beschränkt) berechnet und in *Ax!* index (Entfernung in x-Richtung) und *Ay!* index (Entfernung in y-Richtung) speichert (Prozedur: *Kreiere_Beschleunigungstabelle*)

Wir betrachten z.B. Stern Nr. 4. Er entsteht in (RND(20)-,RND(20)). Diese Koordinate wird in *X(4)* und *Y(4)* gespeichert (Zeile 25, 26). Gleichzeitig wird ihm eine Anfangsgeschwindigkeit *(Vx!(4),Vy!(4))* zugeteilt, die aus (2+RND(2), 2+RND(2)) errechnet wird (Zeile 27, 28). Seine nächste Bewegung wird folgendermaßen charakterisiert (wir betrachten nur die x-Richtung, in Y-Richtung passiert dasselbe): Zu seiner Geschwindigkeit Vx!(4) wird die Beschleunigung *Ax!(x(4))* addiert. Dann wird zu seiner Koordinate *x(4)* die Geschwindigkeit *Vx!(4)* addiert. Das ergibt seine neue Koordinate *x(4)*, an der dann der Stern im nächsten Bild erscheint. In y-Richtung geschieht dasselbe. Die Prozedur *Berechne_neue_-Koordinaten* erledigt das (Zeilen 31 bis 42). Zusätzlich wird hier noch getestet, ob der Stern überhaupt noch innerhalb des Bildschirms liegt. Falls nicht, wird er in der Mitte neu geschaffen.

Die ganze Sache vereinfacht sich und gewinnt enorm an Geschwindigkeit, wenn man das alles für nur ein Viertel des Bildschirms (Quadrant) berechnet und dann symmetrisch auf die restlichen 3 Quadranten überträgt. Das geschieht in Zeilen 49 bis 52. Da 0<= x(i) <= 320 und 0<= y(i) <= 200 gilt (Zeile 38 bis 40), läßt sich das auf diese Art lösen.

Gelöscht wird die alte Grafik in Zeile 47 durch Hinsetzen eines Backgrounds anstelle eines CLS.

Somit sollte das Hintergrundprogrammieren von Grafiken eigentlich klar sein. Diese Technik eröffnet jedem, der sie beherrscht, eine völlig neue Dimension der Programmierkunst. Bewegte Grafiken sind jetzt ein Kinderspiel, ebenso die damit verbundenen Anwendungen.



```
 1:  '*** Beispielprogramm Hintergrundgrafik ***
 2:  '*** in OMIKRON.BASIC, von G. Schabrun   ***
 3:  '*** (c) MAXON Computer GmbH 1990        ***
 4:  CLEAR 100000'...für MEMORY(...) notwendig
 5:  CLIP 0,0,640,400'...nicht vergessen!
 6:  PRINT CHR$(27);"f"'...Cursor ausschalten
 7:  Anzahl_Sterne%L=5'...pro Quadrant
 8:  DIM X%L(Anzahl_Sterne%L),Y%L(Anzahl_Sterne%L)
 9:  DIM Vx!(Anzahl_Sterne%L),Vy!(Anzahl_Sterne%L)
10:  DIM Ax!(320),Ay!(200)
11:  XBIOS (Ad1%L,2)
12:  Ad2%L=256* INT(( MEMORY(32256)+256)/256)
13:  MODE =3'...GfA: GRAPHMODE 3
14:  '
15:  '---------- START --------------
16:  Kreiere_Hintergrund
17:  Kreiere_Beschleunigungstabelle
18:  Zeichne_Zum_Ersten_Mal
19:  REPEAT
20:    Berechne_Neue_Koordinaten
21:    Zeichne_Sterne
22:  UNTIL Weissnichtwann%L=9
23:  '---------- ENDE ---------------
24:  '
25:  DEF PROC Kreiere_Stern(I%L)
26:    X%L(I%L)= RND(20)
27:    Y%L(I%L)= RND(20)
28:    Vx!(I%L)=2+ RND(2)
29:    Vy!(I%L)=2+ RND(2)
30:  RETURN
31:  '
32:  DEF PROC Berechne_Neue_Koordinaten
33:    LOCAL I%L
34:    FOR I%L=0 TO Anzahl_Sterne%L
35:      Vx!(I%L)=Vx!(I%L)+Ax!(X%L(I%L))
36:      Vy!(I%L)=Vy!(I%L)+Ay!(Y%L(I%L))
37:      X%L(I%L)=X%L(I%L)+Vx!(I%L)
38:      Y%L(I%L)=Y%L(I%L)+Vy!(I%L)
39:      IF X%L(I%L)>320 OR Y%L(I%L)>200 THEN
40:        Kreiere_Stern(I%L)
41:      ENDIF
42:    NEXT I%L
43:  RETURN
44:  '
45:  DEF PROC Zeichne_Sterne
46:    LOCAL I%L
47:    Switch_Bildschirm
48:    MEMORY_MOVE Back%L,32000 TO Ad1%L
49:    FOR I%L=0 TO Anzahl_Sterne%L
50:      DRAW 320+X%L(I%L),200+Y%L(I%L)
51:      DRAW 320+X%L(I%L),200-Y%L(I%L)
52:      DRAW 320-X%L(I%L),200+Y%L(I%L)
53:      DRAW 320-X%L(I%L),200-Y%L(I%L)
54:    NEXT I%L
55:  RETURN
56:  '
57:  DEF PROC Switch_Bildschirm
58:    SWAP Ad1%L,Ad2%L
59:    XBIOS (,5,L Ad1%L,L Ad2%L,-1)
60:  RETURN
61:  '
62:  DEF PROC Zeichne_Zum_Ersten_Mal
63:    FOR I%L=0 TO Anzahl_Sterne%L
64:      Kreiere_Stern(I%L)
65:      FOR J%L=0 TO RND(5)
66:        Berechne_Neue_Koordinaten
67:        Zeichne_Sterne
68:      NEXT J%L
69:    NEXT I%L
70:  RETURN
71:  '
72:  DEF PROC Kreiere_Hintergrund
73:    FILL STYLE =2,8
74:    PBOX 0,0 TO 639,399: BOX 4,4 TO 635,395
75:    Back%L=256* INT(( MEMORY(32256)+256)/256)
76:    MEMORY_MOVEB Ad1%L,32000 TO Back%L
77:  RETURN
78:  '
79:  DEF PROC Kreiere_Beschleunigungstabelle
80:    FOR X%L=0 TO 320
81:      Ax!(X%L)=(X%L^2.5/20000)
82:    NEXT X%L
83:    FOR Y%L=0 TO 200
84:      Ay!(Y%L)=(Y%L^2.5/30000)
85:    NEXT Y%L
86:  RETURN
```



Ladenlokal: Gustav-Mahler-Straße am Einkaufszentrum
viele Parkplätze direkt am Laden

Weide Elektronik GmbH
Regerstr. 34
D-4010 Hilden
Telefon 0 21 03 / 4 12 26
Telefax 0 21 03 / 3 18 20

ATARI System-fachhändler
Leasing für Komplettsysteme möglich
Atari + PC XT/AT

Schweiz
DTZ Data Trade AG
Landstr. 1
CH-5415 Rieden - Baden
Tel. (056) 82 18 80
Fax (056) 82 18 84

Österreich
Reinhart Temmel GmbH
Markt 109
A-5440 Golling
Tel. (0 62 44) 70 81-0
Telefax 7 18 83

STECHUHR ST

Manipulationsgeschützte Software zur Erfassung von Arbeitszeiten mittels Kennungsdiskette für jeden Mitarbeiter

Nutzen

- verläßliche Stundenlisten
- individuelle Lohngestaltung und Lohnabrechnung
- Kostensenkung durch Zeitersparnis und exaktes Datenmaterial

Leistungsmerkmale

- Sekundengenau erfaßte Ankunfts- und Abgangszeiten
- einfache Erfassung durch Einschieben der für jeden Mitarbeiter spezifischen Kennungs-Diskette
- Entnahme der Protokolldiskette akustisch gesichert
- Datum und Uhrzeit werden automatisch über die interne Uhr des Rechners erfaßt
- exakt berechnete Arbeitsstunden pro Mitarbeiter
- beliebig wählbare Auswertungszeiträume
- Protokoll der täglichen Arbeitszeiten auf Diskette und Drucker (optional)
- automatische Fehlerkorrektur bei vergessener Erfassung
- MS-DOS kompatibles Diskettenformat
- der Rechner ist weiterhin für andere Zwecke einsetzbar, falls als ACC benutzt
- Zeiten können für Lohnabrechnung gewichtet werden
- individuelle Lohnabrechnung für jeden Mitarbeiter durch
- eigene Abrechnungsformel mit zeit- und lohnabhängigen Faktoren
- komplexe Berechnungen in Abrechnungsformularen

ADRESS DIREKT

Datensicherheit

- geschützt durch Passwort
- Schutz der Installation vor Manipulationen wie Löschen etc.
- ‚private' Datensätze speicherbar, d.h. diese sind nur für den Anwender zugänglich, der sie gespeichert hat - Ausnahme: der Superuser kann alle sehen
- beliebig viele Installationen speicherbar
- Zugriffsrechte für ‚Speichern', ‚Ändern', ‚Löschen', ‚private Datensätze', ‚Drucken', ‚Listenbearbeitung', ‚Installationserlaubnis' einzeln und für jeden Benutzer getrennt einstellbar
- jeder Installation kann ein eigener Pfad für den Dateizugriff vergeben werden
- Adressdaten nicht in einer Datei, sondern in mehreren verteilt, dadurch bei Diebstahl nicht reproduzierbar
- viele Einzeldaten kodiert abgespeichert

Komfort

- jeder Installation, das heißt jedem Benutzer, kann eine eigene Textverarbeitung zugeordnet werden
- Start beliebiger Fremdprogramme möglich
- bis zu 7 Fenster mit beliebigen Inhalten
- großbildschirmfähig
- Zugriff auf die installierten ACC's
- bequemes Fensterhandling auch über Tastatur
- weit über den GEM-Standard hinausgehende Möglichkeiten der Dialogbearbeitung
- Popup-Menüs mit den wichtigsten Funktionen
- sofortiges Erkennen von Falscheingaben
- sofortige Feldformatierungen mit intelligenten Ergänzungsroutinen für Datum/Zeit
- editierbare Dateilisten
- in Listen geänderte bzw. gelöschte DS werden hervorgehoben
- Dateilisten abspeicherbar und wieder ladbar (stets mit aktuellen Werten!)
- schnelle Textsuche auch in Dateilisten unter Verwendung von Wildcards
- mächtige Listenfunktionen mit Rechnen im Text,
- Platzhalter für Datum, Uhrzeit, vielseitige Formatiermöglichkeiten
- verschachtelte Listenausgabe möglich
- schnelle Listenausgabe auf Laserdrucker, bei extrabreiten Listen auch gedreht
- flexibler Import (z.B. von BS-Handel-Daten, 1st-Address, Adimens, DBase)

Netzwerkfähige Version für das PAM's Net lieferbar

COMPUTER-PROJEKTIONSPANEL

Für alle, die etwas präsentieren müssen!

- Gestochen scharf, 8 Graustufen, hochmodernes DST
- LCD mit IR Fernbedienung!
- Keine Spezialsoftware nötig.
- Geeignet für jeden Overhead Projektor.
- Invertier-Funktion, Löschfunktion, auch an IBM PS/2, IBM PC XT/AT mit CGA, EGA, VGA-Karte anschließbar!
- Auflös. 640 x 480 Bildpunkte, nur 3 kg Gewicht!

Preis 3798,-

Passender Metallkoffer 348,-

SOFTWARE

CAD	
Cadja	998,-
Technobox Drafter	798,-
Technobox CAD/2	1998,-
Vorführung von kompletten CAD-Syste-men unter MS-DOS oder auf Atari ST in unserem Beratungszentrum. Rufen Sie an!	
DTP	
Calamus	798,-
Calamus Font Editor	198,-
Outline Art	378,-
PKS-Write	198,-
Didot Fonteditor f. Calamus	198,-
Wir bieten Ihnen als DTP-Center indivi-duelle Beratung für den Systemkauf im Bereich Desktop-Publishing. Rufen Sie an!	
Textverarbeitung	
1st Word Plus V3.15 inkl. 1st Address u. 1st Extra	239,-
1st Proportional Plus	119,-
Signum II	438,-
Scarabus Signum Fonteditor	98,-
Daily Mail	179,-
Writer ST	148,-
Word Perfect ST	748,-
That's Write Junior	148,-
That's Write	328,-
That's Write PostScript	498,-
Tempus 2.0	129,-
Tempus Word	a.A.
Graphik	
Creator	229,-
Imagic	458,-
Megapaint II	449,-
Megapaint II prof.	798,-
Omikron DRAW! 3.0	119,-
STAD	169,-
Retouche	399,-
Arabesque	268,-
Lavadraw	149,-
Datenbanken	
1st Address	148,-
Adimen's	389,-
dbMAN V Version 5.2 incl. Comp., ISI.+Tempus	978,-
DFÜ	
BTX-Manager incl. Interface	398,-
ohne Interface	278,-
Interlink ST	79,-
Tabellenkalkulation	
BasiCalc	98,-
BasiChart	198,-
JAMES Das Börsenprogramm	298,-
GFA-Statistik	948,-
Kaufm. Software	
Software BS-FiBu	548,-
Software BS-Handel	498,-
Software BSS Plus System	a.A.
Software BSS Plus System (netzwerkfähig GTI-Netz)	a.A.
Programmiersprachen	
GFA-ASSEMBLER ST	139,-
Omikron Assembler	94,-
Omikron Macro-Assembler	
GFA-Basic 2.0 EWS ST	44,90
GFA-Basic 3.0 EWS ST	189,-
GFA-Basic 3.5 EWS ST 40 neue Befehle (Vektor/Matrix)	268,-
GFA-Basic 68881 EWS ST	268,-
Omikron Basic	14,90
Omikron Basic 68881 Comp.	219,-
Omikron Basic Compiler	169,-
Omikron Basic Libraries	
diverse Libraries	ab 89,-
Omikron Basic Lib. SQL	288,-
Laser C	388,-
Prospero C	248,-
Turbo C	229,-
Turbo C + Debugger	448,-
Prospero Fortran	378,-
Prospero Fortran 68881 Lib.	198,-
SPC Modula-2	348,-
Prospero Pascal	248,-
GFA-BASIC KONVERTER nach C	478,-
GFA-CUP GEM UTILITY-PACKAGE	139,-
MIDI	
Steinberg Cubase	790,-
Steinberg Twelve	99,-
EMC-Soundeditoren für verschied. Synthesizer/ Expander	
Roland D10/20/110	120,-
Ensoniq VFX	150,-
andere	a.A.
Utilities	
Harlekin *ein ACC für alle Fälle	129,-
Turbo ST* Der Softwareblitter macht dem ST Beine	79,-
Querdruck * universell anwendbar für Tabellenkalkulation	79,-
BOOT IT * DAS Startprogramm für Festplatten	69,-

HARDWARE

Atari Mega ST1 mit 4 MB + SM124 + Maus	2998,-
Weide SCSI-Festplatten	
Weide HD 50, 28 ns	1498,-
Weide HD 65, 28 ns	1698,-
Weide HD 85, 28 ns	1898,-
Weide HD 111, 19 ns Cache	2298,-
Weide HD 132, 19 ns Cache	2548,-
Weide HD 172, 19 ns Cache	3198,-
größere Platten	a.A.
Alle Festplatten mit umfangreichen HD-Utilities, 100% komp. zu Atari-Festplatten, stabiles Metallgehäuse, ohne Lüfter!	
Streamer	
ICD-Streamer 155 MB	2498,-
Komfortable, schnelle u. zuverlässige Sicherung Ihrer Daten auf eine Datenkassette. Datensicherung mit 6.5 MB/Minute, durchgeführter und gepufferter DMA- und SCSI-Bus	
Sub-Systeme	
(*HD + Streamer in einem Gehäuse)	
Weide HD 50 + Streamer	3798,-
Weide HD 111 + Streamer	4598,-
Weide HD 132 + Streamer	4998,-
Weide HD 172 + Streamer	5998,-
Floppy	
3.5", anschlußfertig	289,-
5.25", 40/80 Tracks	448,-
Plotter	
DIN-A3, Flachb., 8 Stifte	2451,-
DIN-A4 bis DIN-A0	a.A.
u.a. Graptec A0-Plotter vorführbereit in unserem Beratungszentrum. Rufen Sie an!	
Scanner	
Printtechnik 200 dpi	948,-
Handyscanner 400 dpi, 105 mm	548,-
Bildverarbeitung auf Atari ST	
Epson GT-4000 Farbscanner 8 Bit, professionell	4998,-
Emulatoren	
PC-Speed	498,-
Supercharger	598,-
Spectre	798,-
Spectre GCR	1298,-
Modem	
Discovery 2400 C	448,-
Discovery 2400 P Pocketmodem	449,-
Der Anschluß und Betrieb der Modems am Netz der DBP ist strafbar!	

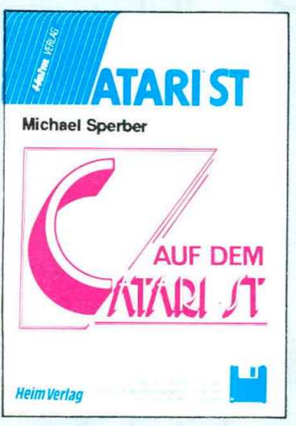
Weide Speicherkarten, die bewährten!

auf 1.0 MB für Atari 260, 520	348,-
auf 2.5 MB für Atari 260, 520, 1040	898,-
auf 2.5 MB für Atari Mega ST1	948,-
auf 4.0 MB für Atari Mega ST1	1548,-
auf 4.0 MB für Atari 260, 520, 1040	1498,-

Jede Erweiterung einzeln im Rechner getestet. Bebil-derte Einbauanleitung, einfacher Einbau ohne Löten. Durch vergoldete Mikrosteckkontakte optimale Schonung des MMU-Sockels. Kein Bildschirmflimmern, keine zus.Software, keine zus. Stromversorgung

Echtzeituhr Plus	129,-
Ab TOS '87 keine Boot-Software nötig. Jede Uhr im Rechner getestet und gestellt. Interner Einbau ohne Löten. Dadurch freier ROM-Port. Schaltjahrerkennung, Datum, Uhrzeit. Tausendfach bewährt!	
Weide Coprozessor 68881	
f. Mega ST incl. Software	498,-
Weide Coprozessor 68881 + WCL	898,-
Weide Coprozessor Language bringt Höchstleistung mit dem 68881 Coproz. INFO anfordern!	
ICD Host Adapter	
für Festplatten im Selbstbau	
Anschluß von SCSI-Festplatten an Atari ST. Voller SCSI-Befehlssatz verfügbar. Umfangreiche Harddisk-Utilities.	
Micro ST, die kompakteste Lösung für den Anschluß einer SCSI-Festplatte	249,-
Advantage ST, mit gepuffertem, durchgeschleiftem DMA-Bus	269,-
Advantage Plus ST mit zusätzl. Echtzeituhr	298,-
Digitizer	
SAM-Echtzeit-Digitizer	998,-
Turbo-Dizer	348,-
Prommer	
GAL-Prommer v. Maxxon	185,-
Junior Prommer	229,-
Rocke Eprommer	239,-
SIMM Speichererw. f. Atari STE	
auf 2.0 MB	598,-
auf 4.0 MB	1198,-
Blitter Nachrüstplatine	89,-
Eprombank 64 KB	19,80
Epromdisk 512 KB	139,-
Atari Trackball f. Atari ST	98,-
kabellose Infrarot-Maus	198,-
Marconi Trackball, d. Professionelle	198,-

NEU & AKTUELL



Buch incl. Programm-Diskette
Hardcover B-406
DM 59,-



Über 570 Seiten
mit Programmdisketten **DM 59,-**
Hardcover B-415



ca. 300 Seiten
Hardcover B-409 DM 49,–
Programmdiskette zum Buch: D-249 DM 39,–



über 530 Seiten
Bestell-Nr. B-419 **54,–**
ISBN 3-923250-69-X
Inclusive Programmdiskette



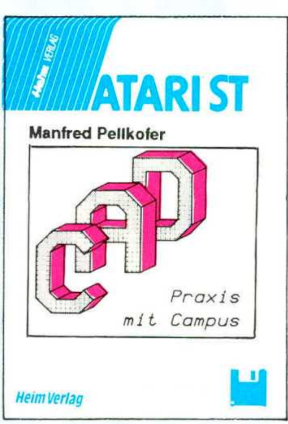
Über 300 Seiten
B-414
DM 49,-



Hardcover, über 430 Seiten
mit Programmdiskette
B-421
DM 69,-



Hardcover, 453 Seiten
mit Programmdiskette **DM 59,-**
B-400



ca. 290 Seiten
Bestell-Nr. B-418 **59,–**
ISBN 3-923250-67-3
Inclusive Programmdiskette



Hardcover
Bestell-Nr. B-436 **DM 59,-**
ISBN 3-923250-77-0
Inclusive Diskette mit Interaktiver Assembler-Entwicklungssoftware



220 Seiten – Hardcover
Bestell-Nr. B-432 **DM 49,-**
ISBN 3-923250-76-2
Inclusive Diskette mit Sicherheitssystem



Hardcover
Bestell-Nr. B-435 **DM 59,-**
ISBN 3-923250-79-7



über 330 Seiten
Bestell-Nr. B-420 **54,–**
ISBN 3-923250-70-3

Alle Preise sind unverbindlich empfohlene Verkaufspreise.

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Bitte senden Sie mir: ______________________

zuzügl. Versandkosten DM 6,– (Ausland DM 10,–) unabhängig von der bestellten Stückzahl

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ______________________

Straße, Hausnr. ______________________

PLZ, Ort ______________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Schweiz
Data Trade AG
Landstr. 1
CH - 5415 Rieden - Baden

Österreich
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

Dekompression



Im letzten Monat haben wir uns mit dem Komprimieren von Daten nach dem Lempel-Ziv-Welsch-Verfahren beschäftigt. In der heutigen Ausgabe der ST-Ecke möchte ich näher auf den Dekomprimieralgorithmus des LZW-Verfahrens eingehen.

Da der Dekomprimierer 'trocken' schlecht zu verstehen ist, soll die Erkärung an einem Beispiel geschehen. Es entspricht dem in der letzten ST-Ecke verschlüsselten Text 'abcabc'. Auch wenn bei diesem Beispiel der Komprimierungsgrad noch nicht sehr hoch ist, zeigt er doch sehr gut die Vorgehensweise des Komprimierers und des Dekomprimierers.

Eingabe : ((int) 'a'), ((int) 'b'), ((int) 'c'), 256, ((int) 'c').

Bezeichnungen:
curr_code: aktueller Code
last_code: vor curr_code bearbeitetes Codewort
code_tab : Codierungstabelle - ein Array, in dem jedes Element 2 Zeichen (char) und einen Index in die Decodierungstabelle speichern kann:

```
typedef struct
{
  char     first, last;
  unsigned prefix;
} decode_elem;
                /* Ein Element der Tabelle */
```

Jeder Eintrag in der Decodierungstabelle bezeichnet im Prinzip einen String, wobei die Einträge mit gleichem Index in der Codierungs- und der Decodierungstabelle dieselben Strings bezeichnen. Allerdings ist die Darstellung in der Decodierungstabelle nicht sofort einsichtig. Sie sieht folgendermaßen aus: Die beiden Zeichen pro Element der Decodierungstabelle geben das erste bzw. letzte Zeichen des Strings an, der durch diesen Eintrag bezeichnet wird (die beiden Zeichen werden im folgenden mit first und last bezeichnet).

Der Präfix

Der Index in die Codierungstabelle kennzeichnet den Präfix des bezeichneten Strings (der Index wird im folgenden mit prefix bezeichnet). Der Präfix eines Strings ist bei Strings, die nur aus einem Zeichen bestehen, nicht vorhanden (NO_PREFIX in LZW), bei Strings, die aus mindestens zwei Zeichen bestehen, ist es der gesamte String abzüglich des letzten Zeichens. Da bei der LZW-Kompression garantiert ist, daß, sobald ein beliebiger String in der Codierungs-/Decodierungstabelle vorkommt, auch sein Präfix in dieser Tabelle erscheint, kann dieser Präfix einfach durch den Tabelleneintrag repräsentiert werden, in dem es gespeichert ist.

Um zu einem beliebigen Tabelleneintrag den bezeichneten String herauszubekommen, geht man folgendermaßen vor: Man läuft von dem Tabelleneintrag los zu dessen Präfix und von dort zum Präfix vom Präfix usw., bis zu einem Eintrag, der kein Präfix hat [dieser bezeichnet einen String, der nur aus einem Zeichen besteht (siehe oben)]. Nun gibt man das Zeichen aus, das als letztes Zeichen dieses einelementigen Strings gespeichert ist und nun das letzte Zeichen des vorangegangenen Eintrages, der logischerweise einen zweielementigen String bezeichnet usw. bis zum Tabelleneintrag, von dem man gestartet ist, und dessen letztes Zeichen man als letztes ausgibt (diese Ausgabe wird in einer etwas trickreicheren Form von der Funktion *write_data* in LZW ausgeführt, nähere Erklärung kommt noch). Der Vorgang wird in Zusammenhang mit dem unten folgenden Beispiel sicherlich verständlicher.

Auspacken

Die Dekomprimierung geht nun folgendermaßen vor sich: Zuerst wird die Decodierungstabelle initialisiert, indem in jedes Array-Element mit Index i zwischen 0 und 255 beide Zeichen (also erstes und letztes Zeichen des bezeichneten Strings) auf (char) i und der Index in die Decodierungstabelle (also der Präfix) auf NO_PREFIX gesetzt wird. Der nächste freie Eintrag ist code_tab[256] (Bitte beachten Sie den Kasten, in dem die einzelnen Variablen der Übersicht wegen schrittweise aufgelistet sind).

Dann wird der erste Code gelesen und der zugehörige String (klarerweise einelementig) ausgegeben (siehe oben), und dieser erste Code wird auch gleich auf

last_code zugewiesen. Nun wird der nächste Code der Eingabe in curr_code gelesen, woraus folgt, daß jetzt curr_code = (int) 'b' und last_code = (int) 'a' ist und bisher "a" ausgegeben wurde (Schritt 1).

Da schon ein Tabellenentrag existiert, dessen Index gleich curr_code ist (das ist der, der das einzelne Zeichen 'b' enthält), wird der zu curr_code gehörende String ("b") ausgegeben und in dem nächsten freien Tabelleneintrag first auf first von code_tab[last_code] ('a'), last auf first von code_tab [curr_code] ('b') und prefix auf last_code gesetzt (Schritt 2).

Nun wird curr_code in last_code geschrieben und der nächste Code der Eingabe in curr_code gelesen. Der aktuelle Zustand ist also curr_code = (int) 'c', last_code= (int) 'b' und code_tab[256] = {'a', 'b', (int) 'a'}, und bisher wurde "ab" ausgegeben (Schritt 3).

Da schon ein Tabelleneintrag existiert, dessen Index gleich curr_code ('c') ist, wird der zu curr_code gehörende String ("c") ausgegeben und in dem nächsten freien Tabelleneintrag first (des neuen Eintrags) auf first von code_tab[last_code] ('b'), last auf first von code_tab[curr_code] ('c') und prefix auf last_code gesetzt (Schritt 4).

Nun wird curr_code in last_code geschrieben und der nächste Code der Eingabe in curr_code gelesen. Der aktuelle Zustand ist also code = 256, last_code= (int) 'c' und code_tab[257] = {'b', 'c', (int) 'b'}, und bisher wurde"abc" ausgegeben (Schritt 5).

Auch hier existiert schon ein Tabelleneintrag, dessen Index gleich curr_code (= 256) ist, allerdings ist dies kein vorbesetzter Eintrag mehr, sondern der, der bei der Decodierung von (int) 'b' entstanden ist. Trotzdem wird mit ihm gleichartig verfahren, nur wird beim Ausgeben von curr_code diesmal ein zweielementiger String, nämlich "ab" ausgegeben. Also ist nach diesem Schritt der aktuelle Zustand: code_tab[258] = {'c', 'a', (int) 'c'}, es wurde bisher "abcab"ausgegeben (Schritt 6) und curr_code = (int) 'c', last_code = 256 (Schritt 7a).

Im letzten Schritt verfahren Sie wieder wie oben, da schon ein Tabelleneintrag existiert, dessen Index gleich curr_code (= (int) 'c') ist (Schritt 7a).

Interessant sind nun die Werte, die in den nächsten freien Tabelleneintrag geschrieben werden. In first (des neuen Tabelleneintrags) wird first von code_tab[last_code] (= {'a','b', (int) 'a'}), also 'a', in last first von code_tab[curr_code] (= {'c', 'c', NO_PREFIX}), also 'c' geschrieben, und prefix wird zu last_code (= 256). Es werden also der zu code_tab[256] gehörende String und das erste Zeichen des zu code_tab[(int) 'c'] gehörenden Strings zu einem neuen String verbunden, der "abc" lautet und dessen Eintrag code_tab[259](= {'a', 'c', (int)256}) heißt, wobei zu bemerken ist, daß der zu code_tab[256] gehörende String "ab" ist (Schritt 7b). Da in dem letzten Schritt der zu curr_code gehörende String ausgegeben wurde, kam insgesamt "abcabc" zur Ausgabe.

Die schrittweise Dekomprimierung

Schrittvariablen

```
0   code_tab[0].first     =(int) 0
    code_tab[0].last      =(int) 0
    code_tab[0].prefix    =NO_PREFIX

    code_tab[255].first   =(int) 255
    code_tab[255].last    =(int) 255
    code_tab[255].prefix  =NO_PREFIX

1   last_code  = (int) 'a'
    curr_code  = (int) 'b'

    Ausgabe: 'a'

2   last_code  = (int) 'a'
    curr_code  = (int) 'b'
    Ausgabe: 'a','b'
    code_tab[256].first   ='a'  (code_tab[(int)'a'].first)
    code_tab[256].last    ='b'  (code_tab[(int)'b'].first)
    code_tab[256].prefix  =(int)'a' (last_code)

3   last_code  = (int) 'b'
    curr_code  = (int) 'c'

4   last_code  = (int) 'b'
    curr_code  = (int) 'c'
    Ausgabe: 'a','b', 'c'
    code_tab[257].first   ='b'  (code_tab[(int)'b'].first)
    code_tab[257].last    ='c'  (code_tab[(int)'c'].first)
    code_tab[257].prefix  =(int)'b' (last_code)

5   last_code  = (int) 'c'
    curr_code  = (int) 256

6   last_code  = (int) 'c'
    curr_code  = (int) 256
    Ausgabe: 'a','b', 'c', "ab"
    code_tab[258].first   = 'c' (code_tab[(int)'a'].first)
    code_tab[258].last    ='a'  (code_tab[(int)256].first)
    code_tab[258].prefix  =(int)'c' (last_code)

7a  last_code  = (int) 256
    curr_code  = (int) 'c'
    Ausgabe: 'a', 'b', 'c', "ab", 'c'

7b  code_tab[258].first   ='a'  (code_tab[(int)256].first)
```

Feinheiten

Empfängt der Decodierer das Spezialzeichen für "Tabelle löschen", so löscht er die Decodierungstabelle und initialisiert sie wie zu Beginn der Decodierung. Außerdem werden gleich zwei Codes gelesen, wobei der erste in last_code gespeichert und der zugehörige String sofort ausgegeben wird. Empfängt der Decodierer das Spezialzeichen für "Bit-Breite vergrößern", liest er alle folgenden Codes mit einem Bit pro Code mehr, auch er fängt bei 9 Bits an.

Verbesserte Ausgabe...

...des zu einem Eintrag der Decodierungstabelle gehörenden Strings: Da man die Verkettung der einzelnen Einträge mittels des prefix-Feldes als einfach verkettete Liste ansehen kann, ist der naheliegendste Ausgabealgorithmus eine Rekursion, die diese Liste entlangläuft und pro Element der Liste einen rekursiven Aufruf durchführt. Beim Beenden dieser Aufrufe kommt nun immer das last-Zeichen des zu einem Aufruf gehörenden Eintrags (Listenelements) zur Ausgabe.

Der enorme Nachteil dieses Verfahrens ist, daß eine Rekursion mit all ihren bekannten Schwächen dazu nötig ist. Nun gibt es allerdings einen Trick, um die Rekursion in eine Iteration (Schleife) umzuwandeln, ohne daß ein expliziter Stack eingeführt werden muß (ein ähnliches Verfahren verwenden einige LISP-Interpreter zur "speicherschonenden" Garbage-Collection). Zuerst wird die Liste von der Wurzel aus bis zu ihrem Ende durchlaufen, dabei werden alle Pointer umgedreht, das heißt, zeigte zuerst a auf b, so zeigt nun b auf a. Am Listenende angekommen, wird die Liste umgekehrt durchlaufen (durch die umgedrehten Pointer ist dies nun ohne Probleme möglich), und dabei werden die Einträge (in unserem Fall last-Zeichen) bearbeitet und die Pointer wieder zurückgedreht. Dies Verfahren wendet die Prozedur 'write_data' an. Übrigens funktioniert das Ganze auch für das rekursive Durchwandern von Baumstrukturen, sofern pro

Knoten des Baumes noch ein Flag geopfert wird.

Das Programm

In den zwei Programmen sind die Kompression und Dekompression nach dem Lempel-Ziv-Welsch-Verfahren implementiert (Funktionen 'encode' und 'decode'), außerdem einige Routinen, die es erlauben, eine Datei zu komprimieren und zu dekomprimieren. (De-)Komprimiert wird jeweils ein Speicherbereich, dessen Adresse und Länge angegeben wird. Die zu (de-)komprimierenden Daten werden in einen zweiten Speicherbereich geschrieben, von dem auch Adresse und Länge angegeben werden. Es ist möglich, statt einer Zieladresse den Wert 0L zu übergeben, dann wird zwar (de-)komprimiert, aber kein Code erzeugt, sondern lediglich die Länge des (nicht erzeugten) Codes geliefert. Es werden alle oben beschriebenen Optimierungen verwendet, und mit den obigen Erklärungen dürfte das Programm auch recht gut zu verstehen sein.

Sollten Sie allerdings auf die Idee kommen, das obige Beispiel in die Tat umzusetzen, um sich das Ergebnis anzuschauen, so dürften Sie zunächst keine große Freude daran haben, da Sie das Ergebnis nicht wiedererkennen werden. Trösten Sie sich, selbst wir haben im ersten Moment gestutzt, als wir erkannten, daß das Ergebnis nicht so aussieht wie erwartet. Das liegt an folgendem: Wie schon erwähnt, beginnt der Komprimierer mit einer Bit-Breite von 9. Die Bytes, hintereinandergeschrieben und in Neuner-Bit-Pakete eingeteilt, müßten nach Adam Riese und Eva Zwerg unseren oben gezeigten Code ergeben. Da aber einige Optimierungen bezüglich Bit-Schieben etc. durchgeführt wurden, verdrehen sich die Bits innerhalb der im Programm verwendeten Wörter. Wird beispielsweise der erste Code gesendet, verbraucht er nur 9 Bits, so daß er erst einmal zurückgehalten wird. Die restlichen 7 Bits werden mit 7 Bits des folgenden Codes aufgefüllt, das Wort jetzt endlich gesendet, und die fehlenden 2 Bits in das folgende Wort geschrieben, worin nur noch 14 Bits Platz haben, was einem Code plus 5 Bits enspricht und so weiter. Durch das spezielle Verarbeiten der Bits liegen diese dann aber nicht mehr nebeneinander, so daß es per Hand relativ aufwendig ist, zu dekodieren. Wenn Sie aber trotzdem einmal ein eigenes Beispiel anschauen wollen, haben wir allerdings einen Tip parat: Setzen Sie die Konstante CODE_WIDTH von 9 auf 16. Dann wird jeder zu sendende Code (nicht gerade speichersparend) sofort als 16-Bit-Code gesendet. Jetzt können Sie sich die Datei anschauen und recht genau interpretieren. Dazu müssen Sie nur wissen, daß die Bytes des Wortes in der Reihenfolge LO/HI-Byte in der Datei liegen und am Anfang dieser Datei ein Kopf von 12 Bytes [8 Bytes für einen Text und 4 Bytes für die Anzahl der Codes (Langwort)] zu finden ist. Außerdem beginnt der erste freie Eintrag nicht wie im Beispiel bei 256, sondern bei 258 (Hex 102), da 256 und 257 als spezielle Codewörter verwendet werden.

Ganz zum Schluß...

...sei noch angemerkt, daß das hier verwendete Verfahren zwar das gleiche ist wie das von ARC, dennoch sind natürlich die Dateiformate nicht identisch. Und nun wollen wir Ihnen noch viel Spaß beim Ein- und Auspacken Ihrer Daten sowie beim Verstehen des Algorithmus' wünschen. Uns jedenfalls hat es Spaß gemacht (wir haben unseren inzwischen Programmiersprachen-'Streit' zur Seite gelegt und programmieren inzwischen alle drei in beiden Sprachen...).

(Manuel Chakravarty/Gabi Keller/SH)

```
 1:  /*  Lempel - Ziv - Welsch - Komprimierung
 2:   *
 3:   *  Teil 2 : Dekomprimierer
 4:   *
 5:   *  Autoren: Gabi Keller und Manuel Chakravarty
               für ST-Computer
 6:   *  System : LaserC von Application Systems
               Heidelberg
 7:   */
 8:
 9:  #include "LZW.H"
10:
11:
12:  /*  Typedefinition der Decodierungstabelle
13:   */
14:
15:  typedef struct {
16:
17:     /*  Indiziert den Prefix, des durch diesen
18:         Eintrag codierten Wortes        */
19:            unsigned        prefix;
20:            char            first,
    /*  Erstes und letztes Zeichen des durch  */
21:                            last;
          /*  diesen Eintrag codierten Worts */
22:
23:          } decode_elem;
24:
25:  typedef decode_elem       *decode_tab;
26:
27:
28:                 /*  Decodieren  */
29:                 /*  ==========  */
30:
31:  /*  init_decode_tab -- Initialisiere die
         Decodierungstabelle
32:   */
33:
34:  void       init_decode_tab (table)
35:
36:    decode_tab  table;
37:
38:    {
39:      unsigned        i;
40:
41:      for (i = 0; i <= 255; i++)
42:      {
43:             table[i].first = (char) i;
44:         table[i].last  = (char) i;
45:         table[i].prefix = NO_PREFIX;
46:      }
47:    }
48:
49:  /*  read_code -- Ließt das nächste Codewort und
50:         setzt 'endOfSource', falls
51:   *  kein Codewort mehr vorhanden ist. */
52:
53:  void       read_code (bit, bit_width, source,
                 source_len, code, end_of_source)
54:
55:    int               *bit, bit_width;
56:    char              **source;
57:    long              *source_len;
58:    unsigned          *code;
59:    bool              *end_of_source;
60:
61:    {
62:
63:      unsigned long          l, lb;
64:      unsigned               i, shift, mask;
65:      int                    bits;
66:
67:      /*  Lies soviele Bytes, wie es für die
           aktuelle Bitbreite erforderlich ist.*/
68:                                             →
69:      bits = *bit + bit_width;
```

```
70:        l = 0L; shift = 0;
71:        do
72:        {
73:          if (*source_len == 0L) {*end_of_source =
                                        TRUE; return;}
74:          else
75:          {
76:            lb = (unsigned long) *(*source)++;
77:            lb <<= shift;
78:            l |= lb;
79:            shift += BITS_PER_CHAR;
80:            (*source_len)--;
81:            bits -= BITS_PER_CHAR;
82:          }
83:        } while (bits > 0);
84:        if (bits != 0) {(*source)--;(*source_len)++;}
85:
86:        /* Maskiere die durch Beschränkung auf
              Bytegrenzen überflüssiger Weise
87:         * gelesenen Bits aus.
88:         */
89:        l >>= *bit;
90:        *bit = (*bit + bit_width) % BITS_PER_CHAR;
91:        l &= ~(-1 << bit_width);
92:        *code = (unsigned) l;
93:    }
94:
95:  /*  write_data -- Schreibt das durch 'code'
         codierte Wort in den Zielbereich.
96:   *  Dazu wird die "Pointer"-Kette
         (aus Array-Indizies) der
97:   *  Buchstaben des durch 'code'
         codierten Wortes zuerst
98:   *  vom letzten Buchstaben bis zum
         ersten durchlaufen und
99:   *  dabei alle "Pointer" umgedreht
         (d.h. zeigte erst A auf
100:  *  B, so zeigt nun B auf A),
         danach werden die Buchstaben
101:  *  entlang der umgedrehten
         "Pointer"-Kette vom ersten zum
102:  *  letzten Buchstaben durchlaufen,
         wobei alle Buchstaben
103:  *  ausgegeben und die "Pointer"
         wieder zurückgedreht
104:  *  werden.
105:  */
106:
107: void     write_data (code, table, dest, dest_len,
                          data_len)
108:
109:    unsigned        code;
110:    decode_tab      table;
111:    char            **dest;
112:    long            *dest_len, *data_len;
113:
114:    {
115:      unsigned    last, next, zws;
116:      bool back;
       /*  'TRUE' ~ Rückwärtslauf aktiv */
117:
118:      back = FALSE;
119:      next = table[code].prefix;
120:      last = NO_PREFIX;
121:      while (! back || (next != NO_PREFIX)) {
122:
123:        if (next == NO_PREFIX)
       /*  Listenende erreicht => zurück laufen   */
124:        {
125:          back = TRUE;
126:          zws = next; next = last; last = zws;
127:        }
128:        else
      /*  Ein Schritt laufen und Zeiger umdrehen   */
129:        {
130:          table[code].prefix = last;
131:          last = code;
132:          code = next;
133:          next = table[code].prefix;
134:        }
135:
136:        /*  Falls grade zurück gelaufen wird, so
              gib das letzte Zeichen der
137:         *  Zeichenkette aus, die von 'code'
              codiert wird.
138:         */
139:        if (back)
140:        {
141:          if (*dest_len != 0L)
142:          {
143:            *(*dest)++ = table[code].last;
144:            (*dest_len)--;
145:          }
146:          (*data_len)++;
147:        }
148:
149:      }/*while*/
150:      table[code].prefix = last;
151:    }
152:
153:
154: /*  decode -- Codiere ein Speicherbereich nach
         dem LZW-Verfahren. Dabei wird
155:  *  der Code in einen zweiten Bereich
         beschrieben (Zielbereich).
156:  *
157:  *  'source'      - Startadresse des zu
                       decodierenden Speicherbereichs
158:  *  'source_len'  - Länge des zu
                       decodierenden Speicherbereichs
159:  *  'dest'        - Startadresse des
                       Zielbereichs
160:  *  'dest_len'    - Länge des Zielbereichs
161:  *  'code_len'    - Liefert die Länge der
                       Daten im Zielbereich
162:  *                 nach Abschluß der Decodierung.
163:  *
164:  *  Wird als Länge oder Startadresse des
         Zielbereichs '0L' ange-
165:  *  geben, so wird die Codierung
         durchgeführt, aber kein Code
166:  *  erzeugt, sondern lediglich die Länge
         der Codedaten ermittelt.
167:  */
168:
169: void     decode (source, source_len, dest,
                     dest_len,data_len)
170:
171:    char    *source;
172:    long    source_len;
173:    char    *dest;
174:    long    dest_len;
175:    long    *data_len;
176:
177:    {
178:
179:      decode_tab    table;        /* Decodierungs-
                                         tabelle */
180:      unsigned      last_code,    /* vorheriges
                                         Codewort   */
181:                    code,         /* aktuelles
                                         Codewort   */
182:                    next_entry;   /* nächster
                                         freier Platz
                                         in 'table' */
183:      int           bit,          /* nächstes Bit
                                         in 'dest'  */
184:                    bit_width;    /* aktuelle Bit-
                                         breite der
                                         Codewörter */
185:      bool end_of_source;  /*  zeigt Ende der
                                   Codedaten an   */
186:
187:      /*  Alloziere den Speicherbereich für die
              Decodierungstabelle.
188:       */
189:      *data_len = 0L;
190:      if (source_len == 0L) return;
191:      table = (decode_tab) Malloc ((long) sizeof
                  (decode_elem) * TABLESIZE);
192:      if (table == NULL) return;
193:      if (dest == NULL) dest_len = 0L;
194:
195:      /*  Belege die Variablen vor
196:       */
197:      bit = 0;
198:      bit_width = 9;
199:      next_entry = FIRST_FREE;
200:
201:      /*  Belege die Decodierungstabelle mit den
              Werten von 0 bis 255 vor. */
```

→


ATARI ST
COMPUTER
Die Fachzeitschrift für den ATARI-ST Anwender.
DM 14,- Ös. 112,- Sfr. 14,-
16 MHz-Tuning im Selbstbau
Mega-Modul
MBit EPROMs selbst gebrannt
DPE
3 Druckerports am ST
PC Speed-Tips
Einführende Grun
DAS GROßE ST - EXTRA.
BESTELL - COUPON EINSENDEN AN: HEIM VERLAG, HEIDELBERGER LANDSTR. 194, 6100 DARMSTADT 13
STÜCK ST - EXTRA ZUM PREIS VON 14,- DM JE HEFT
VERSAND FREI HAUS
ZAHLUNG: 14,- DM ALS SCHECK IM BRIEFUMSCHLAG
DEM BESTELLCUPON BEILEGEN.
NAME
STRAßE
ORT

```
202:
203:      init_decode_tab (table);
204:
205:      /*  Lies das erste zu decodierende Zeichen
             und gib es aus */
206:
207:      end_of_source = FALSE;
208:      read_code (&bit, bit_width, &source,
                  &source_len, &code, &end_of_source);
209:      last_code = code;
210:      write_data (code, table, &dest, &dest_len,
                     data_len);
211:
212:      if (! end_of_source)
213:        read_code (&bit, bit_width, &source,
                     &source_len, &code,
                     &end_of_source);
214:      while (! end_of_source) {
215:
216:        /*  Spezialcode für Erweiterung der
               Bitbreite => Bitbreitenerweiterung
217:         *  Spezialcode für Löschen der Tabelle =>
               Löschen der Decodierungstab.
218:         *  Sonst wird das Wort, das durch den
               aktuellen Code codiert wird
219:         *  in die dekomprimierten Daten
               geschrieben und die Decodierungs-
220:         *  tabelle um ein Wort erweitert.
221:         */
222:        if (code == INCREASE_WIDTH) bit_width++;
223:        else if (code == NEW_TABLE) {
224:
225:          /*  Lösche die Decodierungstabelle und
                 lies das nächsten Zeichen */
226:
227:          bit_width = 9;
228:          next_entry = FIRST_FREE;
229:          read_code (&bit, bit_width, &source,
                       &source_len, &code,
                       &end_of_source);
230:          if (!end_of_source)
231:          {
232:            last_code = code;
233:            write_data (code, table, &dest,
                          &dest_len, data_len);
234:          }
235:
236:        } else  {
237:
238:          /*  Schreibe das zum aktuelle Code
                 gehörende Wort und erweitere
239:           *  die Decodierungstabelle um das Wort,
                 das aus dem zum vorher-
240:           *  gehenden Code gehörenden Wort plus
                 dem Anfangsbuchstaben
241:           *  des zum aktuellen Code gehörenden
                 Wortes besteht.
242:           *  Achtung: Es kann auch passieren, daß
                 'code == next_entry' ist.
243:           */
244:          table[next_entry].prefix = last_code;
245:          table[next_entry].first  =
                   table[last_code].first;
246:          table[next_entry].last   =
                   table[code].first;
247:          next_entry++;
248:          last_code = code;
249:          write_data (code, table, &dest,
                         &dest_len, data_len);
250:
251:        }
252:        read_code (&bit, bit_width, &source,
                     &source_len, &code,
                     &end_of_source);
253:
254:      }/*while*/
255:
256:      Mfree (table); /*  Gib den Speicherplatz der
                         Decodierungstabelle frei  */
257:    }
258:
259:                /*  Benutzer-Schnittstelle  */
260:                /*  ======================  */
261:
262:  /*  get_ch -- Ließt ein Zeichen von standard
         input. */
263:
```

```
264:
265:  void     get_ch (ch)
266:
267:    char   *ch;
268:
269:    {
270:      printf ("\33e"); fflush (stdout);
271:      *ch = (char) Cconin ();
272:      printf ("\33f"); fflush (stdout);
273:    }
274:
275:  /*  err_msg -- Schreibt eine Fehlermeldung
         invertiert in stdout. */
276:
277:
278:  void     err_msg (str)
279:
280:    char   *str;
281:
282:    {
283:      printf ("\33p"); fflush (stdout);
284:      printf ("%s", str);
285:      printf ("\33q"); fflush (stdout);
286:    }
287:
288:  /*  read_file -- Lies angegebene Datei in neu
         allozierten Speicherbreich.
289:   *  'success = TRUE', falls das Laden
         erfolgreich war.                      */
290:
291:
292:  void     read_file (fname, in, in_len, success)
293:
294:    char   fname[], **in;
295:    long   *in_len;
296:    bool   *success;
297:
298:    {
299:      int          f;
300:
301:      *success = FALSE;
302:      f = Fopen (fname, 0);
303:      if (f < 0) {err_msg ("Kann die Eingabedatei
                         nicht öffnen!"); return;}
304:      *in_len = Fseek (0L, f, 2);
305:      Fseek (0L, f, 0);
306:      *in = (char *) Malloc (*in_len);
307:      if (*in == NULL) {err_msg ("Zu wenig
                        Speicherplatz!"); return;}
308:      Fread (f, *in_len, *in);
309:      Fclose (f);
310:      *success = TRUE;
311:    }
312:
313:  /*  write_file -- Schreibt angegebenen
         Speicherbereich in eine Datei.
314:   *  'success = TRUE', falls das Speichern
         erfolgreich war.                       */
315:
316:
317:  void     write_file (fname, out, out_len,
                                       success)
318:
319:    char   *fname, *out;
320:    long   out_len;
321:    bool   *success;
322:
323:    {
324:      int          f, err;
325:
326:      *success = FALSE;
327:      f = Fcreate (fname, 0);
328:      if (f < 0) {err_msg ("Konnte Datei nicht
                   erzeugen!"); return;}
329:      err = Fwrite (f, out_len, out);
330:      Fclose (f);
331:      if (err < 0) {Fdelete (fname); err_msg
                    ("Laufwerk ist voll!");}
332:      *success = TRUE;
333:    }
334:
335:  main (argc, argv)
336:
337:    int    argc;
338:    char   *argv[];
339:
340:    {
```

→

```
341:        char ch, *in, *out;
342:        long in_len, out_len, data_len;
343:        bool success;
344:        int  i;
345:
346:        printf ("Lempel-Ziv-Welsch-
                 Dekomprimierer\n");
347:        printf ("von Gabi Keller und Manuel
                 Chakravarty für die ST-
                 Computer\n\n");
348:        if (argc == 1)
349:        {
350:          printf ("Bitte geben Sie in der
                   Argumentzeile die zu\n");
351:          printf ("dekomprimierenden Dateien an.\n");
352:        }
353:        else
354:        {
355:          for (i = 1; i < argc; i++) {
356:
357:            printf ("%s: lese..", argv[i]); fflush
                     (stdout);
358:            read_file (argv[i], &in, &in_len,
                        &success);
359:            if (success)
360:            {
361:              if (strcmp (in, "LZWComp") != 0)
362:              {
363:                err_msg ("Datei ist nicht
                         komprimiert!");
364:                Mfree (in);
365:                success = FALSE;
366:              }
367:            }
368:            if (success)
369:            {
370:              out_len = *((long *) (in + 8L));
371:              out = (char *) Malloc (out_len);
372:              if (out == NULL)
373:              {
374:                err_msg ("Zu wenig Speicherplatz!");
375:                Mfree (in);
376:                success = FALSE;
377:              }
378:            }
379:            if (success) {
380:
381:              printf ("dekomprimiere.."); fflush
                       (stdout);
382:              decode (in + 12L, in_len - 12L, out,
                       out_len, &data_len);
383:              if (data_len == 0L) err_msg ("Zu wenig
                                 Speicherplatz!");
384:              else if (data_len != out_len)
385:              {
386:                err_msg ("Fehler beim
                         Dekomprimieren!");
387:              }
388:              else
389:              {
390:                printf ("schreibe.."); fflush
                         (stdout);
391:                write_file (argv[i], out, data_len,
                             &success);
392:                if (success) printf ("OK!");
393:              }
394:              Mfree (out);
395:              Mfree (in);
396:
397:            }
398:            printf ("\n");
399:          }/*for*/
400:        }
401:        printf ("\nBitte drücken Sie eine Taste...");
402:        get_ch (&ch);
403:      }
```

Listing 1: Das Listing des Lempel-Ziv-Welsch-Dekomprimierers

```
 1:  /* Lempel - Ziv - Welsch - Komprimierung
 2:   *
 3:   * Header-Datei mit für Komprimierung und
      Dekomprimierung
 4:   * gemeinsam geltenden Definitionen
 5:   *
 6:   * Autoren: Gabi Keller und Manuel Chakravarty
               für ST-Computer
 7:   * System : LaserC von Application Systems
               Heidelberg
 8:   */
 9:
10:  #include <stdio.h>
11:  #include <osbind.h>
12:
13:  #define  BITS_PER_CHAR    8
14:  #define  TRUE             (-1)
15:  #define  FALSE            0
16:  #define NO_PREFIX         ((unsigned) -1)
                              /*  Alle Bits auf 1   */
17:  #define TABLESIZE         1024
18:  #define HASHSIZE 2509    /*  Sollte prim sein  */
19:  #define HASHSTEP 23      /*  Sollte prim sein  */
20:  #define STRSIZE           80
21:  #define INCREASE_WIDTH  256  /*  Spezialcode um
22:  * dem Decodierer mitzuteilen, daP ab dem nächsten
23:  * Codewort die Bitbreite um 1 vergröPert wird */
24:  #define NEW_TABLE        257   /*  ..., daP er
                      seine Tabelle löschen soll */
25:  #define FIRST_FREE       258   /*  Erster Code
                              fuer neue Einträge */
26:
27:  typedef  int      bool;
28:  typedef  char     string[STRSIZE];
```

Listing 2: Der Programmkopf für Komprimierer und Dekomprimierer

Alles Lesens-Werte

Experimente am Schreibtisch

Durch schnelle Computer kann seit geraumer Zeit die Simulation von Naturvorgängen realistisch vorgeführt werden.
Mit diesem Buch werden Sie in die Lage versetzt, in Ihrem ATARI ST Galaxien kollidieren zu lassen, ohne daß der Himmel einstürzt, oder gar die gewagtesten chemischen Experimente zu riskieren, ohne daß gleich das Haus in die Luft gesprengt wird.
Natürlich darf bei allem Spieltrieb die Theorie der Gesetzmäßigkeiten nicht zu kurz kommen. Ausführlich



dokumentierte Listings in GFA-BASIC geben einen Einblick in die programmiertechnische Seite.

Robert Becker
Computersimulationen
337 Seiten und Diskette
DM 59,-
ISBN 3-927065-03-X

Mehr als ein Buch! Mehr als nur Software!

Mittlerweile zählt Scheibenkleister II bereits zu den Standardwerken für den ST. Die Erfolgsautoren Claus Brod und Anton Stepper beschreiben auf fast 900 Seiten alles, was man über Floppies, Festplatten, CD-ROMs und andere Massenspeicher wissen muß. Das Buch beinhaltet einen großen Kursteil, in dem Themen wie Floppy- und Festplattenprogrammierung (über Betriebssystem und direkte Controllerprogrammierung), Aufzeichnungsverfahren, Anschluß von Fremdlaufwerken und sogar Justierung und Reparaturhinweise von Laufwerken u.v.m. Zusätzlich wird eine Diskette mit 1.2 MB Software mitgeliefert, die keine Wünsche offen läßt:
Track- und Sektormonitor, bootfähige RAM-Disk, Hyperformat bringt bis zu 950 kByte, Festplattentreiber CBHD.SYS (Laden von Accessories von beliebigem Laufwerk) u.v.m.



Brod/Stepper
Scheibenkleister II
872 Seiten
und Diskette
DM 79,-
ISBN 3-927065-00-5

Fürs Programmieren unentbehrlich

Die Art und Weise wie benutzererstellte Daten behandelt werden, ob beim Speichern, beim Sortieren oder beim Suchen, ist ein zentrales Thema in der Informatik. Grund genug für Dirk Brockhaus, den verschiedenen Strukturen und Algorithmen auf mehr als 400 Seiten auf den Leib zu rücken.
Angefangen mit den Grundlagen, wie einfachste Strukturelemente, über komplexe Zahlen, sowie verschiedene Methoden für Suchen und Sortieren, bis hin zur Verarbeitung großer Datenmengen umspannt dieses Buch den gesamten Themenbereich.

Dirk Brockhaus
Datenstrukturen
403 Seiten und Diskette
DM 59,-
ISBN 3-927065-02-1

Das Buch zum Handbuch

OMIKRON.BASIC, mittlerweile als ATARI-BASIC bei jedem ST beigefügt, wird in diesem Buch ausführlich beschrieben.
Eine kurze Anleitung für den Neuling führt in die grundlegenden Kennisse der BASIC-Programmierung ein.
Es folgt die Beschreibung der Befehle, Programmierkniffe mit vielen Beispielen, sowie Aufgaben und Lösungen.
Sprites werden erklärt, die Overlaytechnik zeigt das Auslagern langer Programmteile nebst Laden bei Gebrauch.
Was sind Bibliotheken (Libraries)? Das Mitbenutzen fremder Bibliotheksfunktionen, Aufrufe von TOS und GEM, sowie GDOS wird kein Geheimnis bleiben.



Ein weiterer Schwerpunkt ist die Sound- und Grafikprogrammierung unter OMIKRON-Basic, mit Klarheit über Metafiles, IMG-Format usw. Als Spezialität folgt die Beschreibung von internen Multitasking-Befehlen.

Clemens Hoffmann
OMIKRON.BASIC 3.0
355 Seiten
und Diskette
DM 59,-
ISBN 3-927065-01-3



Bestellcoupon MAXON Computer GmbH Schwalbacherstr. 52 6236 Eschborn Tel.: 06196/481811

Name/Vorname

Straße

Ort

Unterschrift

Hiermit bestelle ich:

☐ SCHEIBENKLEISTER II DM 79,00
☐ COMPUTER-SIMULATIONEN DM 59,00
☐ OMIKRON.BASIC 3.0 DM 59,00
☐ DATENSTRUKTUREN DM 59,00

Versandkosten: Inland DM 7,50
Ausland DM 10,00

Auslandbestellungen **nur** gegen Vorauskasse
Nachnahme zuzgl. DM 4,00 Nachnahmegebühr.

☐ Vorauskasse
☐ Nachnahme

Vertrieb in der Schweiz: DTZ DataTrade AG Langstrasse 94 Postfach 413 CH-8021 Zürich Tel.: 01/242 80 88 Fax.: 01/291 05 07
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3
Vertrieb in Frankreich: AROBACE 2 Rue Piemontesis F-75018 Paris Tel: 1/42235044 Fax 1/42545631



Somewhere over the rainbow

Programmverwaltung

Heute geht es munter weiter in der Serie über das GEMDOS des TOS' 1.4. Diesmal sind die Programmverwaltung, der GEMDOS-Kern, die Timer-Funktionen und die zeichenorientierte Ein-/Ausgabe an der Reihe.

Nachtrag zur Speicherverwaltung

Schon ist eine Ergänzung zur letzten Folge fällig geworden.

Ich hatte vor einem Monat zwei neue Fehler in der internen Speicherverwaltung erläutert. Zum schwererwiegenden, der zum Verlust der Zuordnung des Besitzers zu einem Speicherblock führen kann, gibt es seit Januar 1990 ein offizielles Patch-Programm von ATARI. Und das, wie schon letztes Mal gesagt, obwohl ATARI seit August 1988 über diesen Fehler informiert ist. Davon mag jeder halten, was er will, aber es läßt wenigstens darauf hoffen, daß er in zukünftigen TOS-Versionen beseitigt sein wird. Dieses Patch-Programm wurde ebenfalls "gepatched", und zwar von Claus Brod, der den Lesern dieser Zeitschrift wohl ein Begriff sein dürfte, da das XBRA-Verfahren nicht vollständig implementiert war.

Programmverwaltung

Im Anschluß an die Besprechung der Speicherverwaltung bietet es sich an, über die Programmverwaltung zu reden. Im Gegensatz zur Speicherverwaltung gab es hier Änderungen, die direkt für den Anwender und Programmierer sichtbar sind. An der Programm-Hierarchie ("parent"- und "child"-Prozesse), der Aufteilung des Programmspeichers in Segmente (TEXT, DATA, BSS), dem Format des Prozeß-Descriptors (auch "Basepage" genannt) hat sich nichts geändert, daher sei hier auf den entsprechenden Artikel über das alte GEMDOS ([1]) verwiesen.

Erweiterter Programm-Header

Aber schon der "Programm-Header" (PH), mit dem jede Programmdatei beginnt, hat sich geändert (Abb. 1). Das Flag *'ph_flag'*, das von ATARI meines Wissens nach bisher nicht dokumentiert wurde, hat eine leicht geänderte Bedeutung. Das bisher nicht benutzte Feld *'ph_res2'* wird nun als Bit-Maske genutzt, um bestimmte "Feinheiten" beim Laden des Programms festzulegen. Bisher sorgte ein Wert von *'ph_flag'* ungleich Null dafür, daß das Relozieren des Programms unterblieb und das BSS-Segment nicht gelöscht wurde. Außerdem wurde die Programmdatei nicht geschlossen, was aber wohl nicht beabsichtigt war. Nun wird nur noch das Relozieren unterbunden; das Schließen der Datei und das Löschen des Speichers findet genau wie bei gelöschtem *'ph_flag'* statt. Und das Löschen des Speichers ist nun genau der Punkt, wo *'ph_res2'* ansetzt. Bekanntlich löscht GEMDOS nach dem Laden eines Programms nicht nur das BSS-Segment, sondern auch den gesamten dem Programm für Heap und Stack zur Verfügung gestellten Speicherplatz. Ab TOS 1.2 ist diese Löschroutine zwar sehr schnell, doch führt dies schon bei 4 MB RAM zu deutlich merkbaren Verzögerungszeiten beim Laden. Bei gesetztem Bit 0 von *'ph_res2'* wird nun nur noch das BSS-Segment gelöscht, bei gelöschtem Bit 0 ist alles wie bisher. Alle Programme, die also nicht fälschlicherweise annehmen, ihr gesamter Speicher sei gelöscht, können durch Setzen dieses Bits zu einem deutlich schnelleren Ladevorgang veranlaßt werden. Dies muß man natürlich bei jedem Programm einzeln ausprobieren.

ATARI hat hierzu sogar ein eigenes Programm namens MAKEFAST entwickelt, wodurch das Setzen und Löschen dieses Bits ermöglicht wird. Wenn man es auf den AUTO-Ordner und die Accessories losläßt, beschleunigt sich der Boot-Vorgang des Rechners enorm. MAKEFAST hat allerdings seine Tücken, so daß

ich es z.B. nur auf Programme losgelassen habe, von denen ich weiß oder hoffe, daß sie dafür geeignet sind. Trotzdem treten ab und zu unerklärliche Abstürze auf, die nicht auf ein bestimmtes Programm beschränkt sind oder reproduzierbar wären. Da es am GEMDOS meiner Meinung nach nicht liegen kann, muß wohl doch irgendein Programm daran schuld sein. Doch einfach zu ermitteln ist es nicht. Ähnliche Erfahrungen haben auch schon andere Leute gemacht. Auch das MAKE-FAST-Programm selbst ist nicht ganz fehlerfrei. Wenn man ihm mehrere Dateinamen übergibt, ignoriert es offensichtlich den ersten. Bei nur einem Dateinamen klappt dagegen alles bestens.

Die anderen Bits von *'ph_res2'* sind für ähnliche Zwecke reserviert, haben unter TOS 1.4 (und TOS 1.6) jedoch noch keine Bedeutung. Beim TOS 3.0 des ATARI TT werden z.B. die Bits 1 und 2 noch benutzt. Sie haben aber mit der geänderten Speicherverwaltung des TT zu tun, auf die ich erst zurückkommen möchte, wenn das TT-TOS offiziell wird.

Neuer 'Pexec'-Modus

Die GEMDOS-Funktion *'Pexec'* hat einen neuen Modus mit der Nummer 6 bekommen und arbeitet ähnlich wie der Modus 4, d.h. ein schon geladenes Programm (*'child'*) wird gestartet. Der Unterschied besteht darin, daß das *'child'* unter seinem eigenen Namen gestartet wird, da der Besitzer (*'m_own'* des Memory-Descriptors) der Speicherblöcke, die das Programm und seine environment strings beinhalten, das *'child'* selbst und nicht sein *'parent'* (wie beim Modus 4) ist. Dieser Modus ist von Vorteil, wenn der Parent-Prozeß die Kontrolle über das *'child'* abgeben möchte. Wenn der *'child'*-Prozeß terminiert, wird z.B. sein Speicher automatisch freigegeben.

Wie schon in [1] erwähnt, werden Programme immer auf dem CPU-Interrupt-Level (IPL) 0 gestartet. Dies wird nun von ATARI auch für die Zukunft garantiert. Normalerweise setzt dann die BIOS-Routine für den *HBLANK (IPL 2)* den IPL auf 3.

Urprozeß-Descriptor

Bei der Initialisierung des GEMDOS wird ein "Ur-PD" aufgesetzt, der als *'parent'* für die später geladenen Programme dient. Im alten GEMDOS wurde der dafür benötigte Platz von der internen Speicherverwaltung bereitgestellt. Da dies nun aufgrund des neuen Konzepts nicht mehr möglich ist, gibt es eine eigene globale Variable hierfür (*'boot_pd'*).

Behobene und nicht behobene Fehler

Einige Bugs in der Programmverwaltung wurden beseitigt, andere dagegen nicht. "Dieser Satz gilt praktisch immer, oder was?! Denn alles, was entsteht, ist wert, daß es zugrunde geht; drum besser wär's, daß nichts entstünde" (Anm. des Autors: Die letzten beiden Sätze entstammen der Feder meines philosophierenden Mitbewohners Rolf; ich wollte eigentlich sagen: Dieser Satz gilt praktisch für das gesamte GEMDOS).

GEMDOS hatte diverse Fehler, so daß beim Laden des Programms etwas schiefgehen konnte. Bei einem BIOS-Fehler (Diskette nicht lesbar o.ä.) wurde unter bestimmten Bedingungen der schon reservierte Speicher für Programm und Environment nicht freigegeben. Wurde das Programmformat nicht akzeptiert oder war nicht genügend Speicherplatz für das BSS-Segment frei, wurde die Programmdatei nicht geschlossen. Beim *'Pexec'*-Modus 3 ("nur laden") wurde in diesem Fall außerdem nicht der schon belegte Speicher freigegeben. Von diesen drei Fehlern wurden nur die ersten beiden beseitigt.

Es besteht immer noch das Problem, daß die Übernahme der Pfade und I/O-Umleitungen vom 'parent' beim Laden und nicht beim Starten des Programms geschieht (s. [2]). Hier hätten leicht weitere *'Pexec'*-Modi eingeführt werden können, wenn schon eine generelle Änderung aus Kompatibilitätsgründen nicht erfolgt.

Modus 3 ist also weiterhin mit äußerster Vorsicht zu genießen. ATARI gibt sogar zu, daß Modus 0 der einzig zuverlässig funktionierende ist, was darauf schließen läßt, daß hierfür offenbar keine Besserung in Sicht ist.

Die Relozierdaten des Programms dürfen nun länger als 32 KB sein.

Beim internen Aufruf von *'Fopen'* in der Lade-Routine fehlte die Angabe der Modus-Parameters. Davon merkte man i.allg. zwar nichts, aber nun ist es korrigiert.

Von der Kommandozeile werden weiterhin nur maximal 125 Bytes (einschließlich des ersten Längen-Bytes) kopiert. Das abschließende Null-Byte eingerechnet, bleiben also die letzten beiden Bytes des Prozeß-Descriptors ungenutzt.

Bei der Terminierung eines Prozesses müssen die Zähler für die Pfad-Handles (*'pthcntx[]'*) aller Pfade, die der Prozeß kennt, dekrementiert werden. Dort wurde nun eine zusätzliche Sicherheitsabfrage eingebaut, die verhindert, daß *'pthcntx[]'* negativ wird. Das entspricht fast meinem Vorschlag aus der ST-Computer 7/88, S. 119, wo das Problem auch näher erläutert ist (es wird allerdings nur auf "ungleich Null" statt "größer Null" abgefragt, was noch sicherer wäre). Andere Merkwürdigkeiten, die man aber - bei gutem Willen - nicht für Bugs halten muß, finden sich im nächsten Abschnitt.

```
typedef struct
{ int ph_branch; /* Magic $601A */
  long ph_tlen;  /* Länge des TEXT-Segments */
  long ph_dlen;  /* Länge des DATA-Segments */
  long ph_blen;  /* Länge des BSS-Segments */
  long ph_slen;  /* Länge der Symbol-Tabelle */
  long ph_res1;  /* Null, reserviert */
  long ph_res2;  /* Flags zum Laden, reserviert */
  int ph_flag;   /* Relozier-Flag (undokumentiert) */
} PH;
```

Abb. 1: Programm-Header

```
PD boot_pd;               /* $5FB2: Ur-PD, Vater
                             aller Dinge */
PD *act_pd;               /* $5622: Zeiger auf aktiven
                             PD */
char prg_stack[3072];     /* $2B8A: GEMDOS-Stack */
```

Abb. 2: GEMDOS-Variablen der Programmverwaltung

Interne Arbeitsweise

Kommen wir nun nach diesem Vorgeplänkel, das größtenteils auch in der TOS 1.4-Dokumentation von ATARI behandelt wird, zur geänderten Arbeitsweise der Programmverwaltung. *'Pexec'* arbeitet zunächst wie bisher. Der Modus 6 unterscheidet sich von Modus 4 erst unmittelbar vor dem Starten des *'childs'*. Dort wird der Besitzer der beiden Speicherblöcke für das *'child'* selbst und sein Environment auf das *'child'* selbst gesetzt. Dies geschieht mit einer neuen Routine der Speicherverwaltung (die ich letztes Mal noch nicht erwähnt hatte), die die Liste der belegten MDs (*'mal'*) nach einem Block durchsucht, der die gewünschte Anfangsadresse hat, und dann einfach *'m_own'* auf den neuen Wert setzt.

Etwas obskur ist dagegen das Anlegen des Environments für den *'child'*-Prozeß. Wie bisher wird die Länge erst einmal auf eine gerade Zahl aufgerundet, was schon überflüssig ist (*'Malloc'* macht dies schon von sich aus). Es werden aber darüber hinaus immer zwei weitere Bytes allo-

ziert. Kopiert wird jedoch nur das Environment selbst, so daß die letzten zwei Bytes (bzw. drei, je nach Rundung) nicht angerührt werden. Das Ganze ist recht merkwürdig programmiert, was vermuten läßt, daß etwas anderes beabsichtigt war (es fragt sich nur was). Positiv zu vermerken ist, daß die zwei nicht benutzten, noch aus dem Digital Research-GEMDOS stammenden Routinen endlich verschwunden sind, sowie daß einige überflüssige Abfragen in *'Pexec'* selbst gestrichen wurden. Die Änderungen in der Routine, die das eigentliche Laden des Programms besorgt, wurden schon erwähnt.

Interessant ist allerdings noch eine Änderung beim residenten Terminieren eines Programms mit *'Ptermres'*. Dabei wurden die MDs aller "eigenen" Speicherblöcke aus der *'mal'*-Liste ausgehängt und freigegeben. Da der Speicher nicht in der *'mfl'*-Liste eingetragen wurde, war er der GEMDOS-Speicherverwaltung nunmehr unbekannt. Nun werden die MDs zwar noch ausgehängt, aber selbst nicht freigegeben. Einen tieferen Sinn kann ich darin nicht entdecken, da die MDs von GEMDOS nicht mehr benutzt werden, aber weiterhin internen Speicherplatz belegen. Denkbar wäre es, daß eine spätere GEMDOS-Version in der Lage ist, diese MDs wiederzufinden, indem sie die Liste der internen Memory-Descriptoren (IMDs) durchgeht, um diese Blöcke dann endgültig freizugeben oder erneut in die 'mal' einzuhängen. Eine Zusammenstellung der globalen Variablen mit den für das deutsche ROM-TOS 1.4 notwendigen Adressen findet sich in Abb. 2.

GEMDOS-Kern

Im zweiten Abschnitt dieser Folge behandele ich die innersten, teilweise maschinenabhängigen Teile des GEMDOS, wie sie schon in [3] beschrieben wurden. Die globalen Variablen führt Abb. 3 auf.

```
long pcsav;        /* $0FF2: PC bei (X)BIOS-Aufruf (TRAP13,14) */
int srsav;         /* $0FFA: Zwischenspeicher für SR */
long otrap2;       /* $1818: trap2-Vektor vor GEMDOS-Init. */
long sjmp1[3];     /* $5F36: Puffer für setjmp/longjmp */
long sjmp2[3];     /* $5684: Puffer für setjmp/longjmp */
long tldiv;        /* $52C4: von 'ldiv' benutzt */
char ncall;        /* $5E90: Zähler für GEMDOS-Aufrufe */
long dsk_err;      /* $5324: BIOS-Diskerror für GEMDOS-Fehlerbehandlung */
int err_drv;       /* $5FB0: Drive des BIOS-Diskerror */
int err_cnt;       /* $52C8: Versuche bei Disk-Fehler */
long err_stack = 0x1810L;
                   /* $FE8580: Adresse Stack für TRAP2-Error */
char std_devx[]= {-1,-1,-2,-3,-1,-1 };
                   /* Standard-Devices für Kanäle */
DOS_F dos_fx[] /* $FE859C: GEMDOS-Funktionen mit Parameter-Flag */
={{ p_term, 0 },
 { c_conin, 0x80 },
 { c_conout, 0x81 },
 { c_auxin, 0x82 },
 { c_auxout, 0x82 },
 { c_prnout, 0x83 },
 { c_rawio, 0x80 },
 { c_rawcin, 0x80 },
 { c_necin, 0x80 },
 { c_conws, 0x81 },
 { c_conrs, 0x80 },
 { c_conis, 0x80 },
 { ill_func, 0 },
 { ill_func, 0 },
 { d_setdrv, 0 },
 { ill_func, 0 },
 { c_conos, 0x81 },
 { c_prnos, 0x83 },
 { c_auxis, 0x82 },
 { c_auxos, 0x82 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { d_getdrv, 0 },
 { f_setdta, 1 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { t_getdate, 0 },
 { t_setdate, 0 },
 { t_gettime, 0 },
 { t_settime, 0 },
 { ill_func, 0 },
 { f_getdta, 0 },
 { s_version, 0 },
 { p_termres, 1 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { d_free, 1 },
 { ill_func, 0 },
 { ill_func, 0 },
 { d_create, 1 },
 { d_delete, 1 },
 { d_setpath, 1 },
 { f_create, 1 },
 { f_open, 1 },
 { f_close, 0 },
 { f_read, 0x82 },
 { f_write, 0x82 },
 { f_delete, 1 },
 { f_seek, 0x81 },
 { f_attrib, 1 },
 { ill_func, 0 },
 { f_dup, 0 },
 { f_force, 0 },
 { d_getpath, 1 },
 { m_malloc, 1 },
 { m_free, 1 },
 { m_shrink, 2 },
 { p_exec, 3 },
 { p_term, 0 },
 { ill_func, 0 },
 { f_sfirst, 1 },
 { f_snext, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { ill_func, 0 },
 { f_rename, 2 },
 { f_datime, 1 }};
```

Abb.3: GEMDOS-Variablen für GEMDOS-Kern

Initialisierung, TRAP-Handler und Superfunktion

Die Initialisierung des GEMDOS beim Warmstart ist bis auf die andere Initialisierung der internen Speicherverwaltung und einer neuen globalen Variablen gleich geblieben (s. unten). Außerdem werden die Kanäle des "Ur-PD" nun mit den Devices aus der Tabelle *'std_devx'* (s. Abb. 3 und weiter unten) vorbesetzt. Der TRAP1-Handler und die Super-Funktion sind völlig unverändert. Damit sind auch die beiden Fehler der *'Super'*-Funktion noch vorhanden. Der eine wurde jedoch dadurch wegdefiniert, daß es laut ATARI inzwischen offiziell so ist, daß bei einem Parameter 1L der Wert -1L zurückgegeben wird, wenn der Supervisor-Mode aktiv ist (und nicht umgekehrt, wie ursprünglich vorgesehen). Der andere Fehler resultiert in einer Zerstörung des User-Stacks, siehe hierzu [3].

Auch der Aufruf des Dispatchers, anschließend der GEMDOS-Funktionen und die Verwendung eines GEMDOS-eigenen Supervisor-Stacks sind so geblieben. Damit ist GEMDOS weder "re-entrant", noch sind die Probleme mit dem Überschreiben dieses Stacks gelöst.

Auch der TRAP2-Handler ist identisch mit dem des "alten" GEMDOS. Allerdings hatte ich vergessen, ein Feature in [3] zu erwähnen. Wird nämlich im Register D0.W der Wert -1 übergeben, erhält man in D0.L die Adresse des "VDI entry points", also der Routine, die vom GEMDOS normalerweise sowieso angesprungen wird, wenn es mit dem TRAP2-Aufruf nichts anfangen kann.

Der Dispatcher

Den Dispatcher, der das Bindeglied zwischen TRAP-Aufruf und eigentlichen GEMDOS-Funktion dargestellt, habe ich in [3] als zu lang und unübersichtlich bezeichnet. Nun, zumindest zwei Routinen sind jetzt aus diesem Monster herausgezogen - für den Anfang schon nicht schlecht.

Es gibt eine neue globale Variable *'ncall'* (Byte), die alle Aufrufe des Dispatchers mitzählt. Ein Byte reicht hierfür zwar nicht aus, um eine eindeutige Zuordnung zu erreichen, doch wird *'ncall'* nur dazu benutzt, einen GEMDOS-Aufruf vom nächsten zu unterscheiden. Dies wird, wie in der letzten Ausgabe auf S. 136 erklärt, bei der internen Speicherverwaltung benötigt. Bei der GEMDOS-Initialisierung wird *'ncall'* auf Null gesetzt. Im Dispatcher findet auch ein Großteil der I/O-Umleitung statt. Hier hat sich einiges verändert, doch werde ich dies bei den entsprechenden GEMDOS-Funktionen unterbringen.

Der Fehler, der dazu führte, daß Disk-Strukturen bei einem Mediumwechsel nicht freigegeben wurden (s. [3]), ist verschwunden. Das wurde dadurch erreicht, daß der betreffende Parameter keine *'register'*-Variable mehr ist. Da der Fehler offenbar durch einen Compiler-Bug verursacht wurde, werden jetzt alle Parameter bei Funktionsdefinitionen nicht mehr als *'register'* deklariert. Ansonsten ist hier alles beim alten geblieben, einschließlich einiger überflüssiger Anweisungen.

Timer-Funktionen

Bei den GEMDOS-Funktionen für Uhrzeit und Datum, die ursprünglich in [2] erklärt wurden, hat sich nicht viel geändert. Die Interrupt-Routine, die die GEMDOS-Uhr steuert, ist so geblieben, wie sie war, einschließlich einiger redundanter Abfragen. Für die GEMDOS-Funktionen selbst gilt ähnliches, allerdings werden Uhrzeit und Datum bei der Rückgabe in ein *'unsigned int'* verwandelt (das obere Wort ist nun stets Null). Im Gegensatz zur Beschreibung in [2] findet also eine "Nullerweiterung" statt "Vorzeichenerweiterung" auf *'long'* statt. Unverständlicherweise wurde dies nicht konsequent durchgehalten, da die globalen Variablen und die Parameter der Timer-Funktionen weiterhin *'signed int'* sind, aber an allen entscheidenen Stellen der "Cast" zum *'unsigned int'* gemacht wird. Ein Wert von Null für den "Tag" innerhalb der Datumsangabe wird weiterhin nicht als Fehler erkannt.

```
long otimer;   /* $1814: 'etv_timer'-Vektor vor GEMDOS-Init. */
long dos_ms;   /* $5FAC: ms seit GEMDOS-Initialisierung */
int last_ms;   /* $2B88: ms seit letztem Setzen der Uhr */
int dos_date;  /* $60BE: akt. Datum im GEMDOS-Format */
int dos_time;  /* $378A: akt. Zeit im GEMDOS-Format */
int monthl[] = { 0,31,28,31,30,31,30,31,31,30,31,30,31 };
               /* $FE8582 */
```

Abb. 4: GEMDOS-Variablen für Zeitfunktionen

```
#define DEV 3            /* GEMDOS kennt 3 Devices */

long c_buf[DEV][80];  /* $37DC: Eingabepuffer */
long *c_tail[DEV];    /* $60B2: Nächste Schreibposition in c_buf */
long *c_head[DEV];    /* $6110: Nächste Leseposition in c_buf */
char c_cnt[DEV];      /* $5680: Zahl der Zeichen in c_buf */
int c_cmn[DEV];       /* $3BA0: logische Spalte (für TAB) */
```

Abb. 5: GEMDOS-Variablen für Device-Ein-/Ausgabe

Zeichenfunktionen bei Dateien	bis TOS 1.2	ab TOS 1.4
Cconin,... (Zeicheneingabe außer Crawio)	(+)	(+)
Crawio (Eingabe)	(+)	+
Cconout,... (Zeichenausgabe)	-	(+)
Cconws	(+)	+
Cconrs	+	+
Cconis,Cauxis	-	+
Cconos,Cprnos,Cauxos	(+)	+

+ funktioniert offenbar
(+) funktioniert mit Einschränkungen
- funktioniert überhaupt nicht

Abb. 6 - Funktionsfähigkeit der I/O-Umleitung

Wie schon im ersten Teil dieser Serie erwähnt, wird aufgrund eines Programmierfehlers die resetfeste Tastaturprozessor-Uhr nicht erkannt, um damit die GEMDOS-Uhr zu initialisieren, falls keine Echtzeituhr vorhanden ist. Die Adressen der globalen Variablen für das TOS 1.4 sind in Abb. 4 wiedergegeben.

Zeichenorientierte Ein-/Ausgabe

Hier geht es um die Arbeitsweise der mit 'C' beginnenden GEMDOS-Funktionen, und zwar auch unter Berücksichtigung der I/O-Umleitung. Dieser letzte Abschnitt nimmt demnach Bezug auf [4] und [5]. Generell kann man sagen, daß hier eine ganze Reihe von Kleinigkeiten geändert wurden, die sich in der Summe aber deutlich bemerkbar machen.

Die nun folgenden Absätze gelten für auf Devices operierende Felder, also nicht für Dateien. Danach wird die I/O-Umleitung miteinbezogen.

Die Kanäle 4 und 5, die bisher undefiniert waren, werden jetzt wie schon die Kanäle 1 und 0 standardmäßig der Konsole (GEMDOS-Device Handle -1) zugeordnet. Es gibt zwar keine GEMDOS-Funktionen, die direkt diese beiden Kanäle ansprechen, doch können sie mittels *'Fread'*, *'Fwrite'* sowie I/O-Umleitung benutzt werden.

Die Vertauschung der *'Bcostat'*-Aufrufe für MIDI und IKBD ist nun offiziell von ATARI bestätigt und soll gültig bleiben.

Ausgabekontrolle mit Cprnout'

'Bconout' war bisher als *'void'* dokumentiert, lieferte aber in Wirklichkeit den Wert Null zurück, wenn das Zeichen ausgegeben werden konnte, ansonsten einen Wert ungleich Null. Dies aber nur für den Drukker (Device 0),und dann auch nur bei der Centronics-Schnittstelle, also nicht, wenn Device 0 mittels XBIOS-*'Setprt'* der RS232-Schnittstelle zugeordnet wurde. Dies wird nun von ATARI garantiert.

Die GEMDOS-Funktion *'Cprnout'* hingegen gab zufälligerweise den Rückgabewert des zur Ausgabe benutzten *'Bconout'* zurück. Dies wird nun explizit

gemacht und ist von ATARI auch so dokumentiert. Der Sinn soll darin liegen, daß man unter GEMDOS nun eine Kontrolle über das Gelingen der Ausgabeoperation bekommt.

Das Problem dieser an sich vernünftigen Maßnahme besteht nun darin, daß der Rückgabewert des BIOS nur bei *'Bconout'* auf die Centronics-Schnittstelle und der des GEMDOS nur bei *'Cprnout'* wohldefiniert ist. Daher darf man hiervon nur Gebrauch machen, wenn man sich sicher ist, daß keine I/O-Umleitung stattgefunden hat. *'Cprnout'* ist also weiterhin *'void'*, wenn auf ein anderes Device umgeleitet, und die anderen GEMDOS-Ausgabefunktionen sind auch dann *'void'*, wenn sie auf den Drucker umgeleitet werden.

Konsequent wäre es gewesen, daß alle BIOS- und GEMDOS-Ausgabefunktionen einen definierten Wert liefern.

Zeichenpufferung

Bei der GEMDOS-eigenen Zeichenpufferung gibt es Erfreuliches zu vermelden. Die Eingabepuffer werden endlich korrekt verwaltet, und zwar so, wie man es erwarten sollte, nämlich zyklisch. Die Head- und Tail-Zeiger werden nun beim Erreichen des Pufferendes automatisch auf den Pufferanfang gesetzt.

Steuerzeichen-erkennung

Die nächste wichtige Verbesserung gibt es bei der Abfrage der CTRL-Codes, die bei einigen Ein- bzw. Ausgabefunktionen vorgenommen wird und bisher einige Probleme besaß. Zunächst prüft die interne Routine *'c_ctrl'* (s. [4]) das Device nur dann mit *'Bconstat'*, wenn es nicht der Drucker (Device 0) ist. Dadurch wird erreicht, daß bei einer Umleitung von *'Cconout'* auf den Drucker keine CTRL-Codes erkannt werden. Bisher führte dies zu einem Absturz.

Nun werden bis zu 16 Zeichen vom BIOS abgeholt, sofern sie das BIOS schon vom Device gelesen hat (also *'Bconstat'* ungleich 0 liefert), und auf CTRL-Codes überprüft. Diese Maßnahme hat zur Folge, daß GEMDOS wesentlich schneller auf CTRL-Codes reagiert; sie können die eigentlichen Eingabedaten quasi überholen, da sie ausgewertet werden, noch bevor eine Eingabe verlangt wurde. Paare von CTRL-S und CTRL-Q gelten bei dieser Zählung bis 16 als ein Zeichen. Bei einem CTRL-S wird natürlich auf jeden Fall auf das CTRL-Q gewartet, unabhängig davon, ob das BIOS noch Zeichen vorrätig hat.

Falls *'c_ctrl'* von einer Eingaberoutine aus aufgerufen wurde und der GEMDOS-Eingabepuffer hiernach noch leer ist, beginnt das Ganze von vorne, d.h. es wird erneut auf maximal 16 Zeichen gewartet. Das führt dazu, daß CTRL-Codes auch dann erkannt werden, wenn GEMDOS und BIOS noch auf eine Eingabe warten müssen, da die "Wartezeit" nun im GEMDOS und nicht im BIOS verbracht wird. Diese drei Änderungen garantieren nun eine fehlerfreie und flüssige Erkennung der CTRL-Codes im Gegensatz zu dem bisher herrschenden Durcheinander. Damit wurde aber auch ein Nachteil eingehandelt, nämlich ein eher möglicher Überlauf des Eingabepuffers. Das kann passieren, wenn das BIOS wesentlich schneller Daten liefert als das Programm abnimmt. Hierzu sei angenommen, daß ein Programm die Standardausgabe auf die serielle Schnittstelle umgelenkt hat. Es liest nun jeweils einen größeren Daten-Block ein und gibt dann (wenige) Zeichen über *'Cconout'* aus, während das BIOS schon den nächsten Datenblock empfängt. Nach 5 *'Cconout'*-Aufrufen ist der GEMDOS-Eingabepuffer schon voll, da bei jedem 16 Zeichen in den GEMDOS-Puffer gelesen werden.

Dieses Beispiel ist zugegebenermaßen etwas fiktiv, weil so etwas aus Geschwindigkeitsgründen meistens sowieso direkt über das BIOS programmiert wird, aber im Zeitalter der Netzwerke sollte eigentlich jede Ein-/Ausgabeoperation über das GEMDOS laufen.

Sicherer wäre es gewesen, wenn *'c_ctrl'* nur dann so vorausschauend arbeiten würde, wenn "genügend" Platz im Puffer ist.

Korrekte Behandlung von "Sonderzeichen"

Die in [4] beschriebene interne Funktion *'c_out'* wurde leicht verbessert. Beim Mitzählen der Ausgabeposition werden Zeichen mit ASCII-Codes ab 128 nun richtig behandelt, d.h. als druckbare Zeichen angesehen (dieser Fehler war mir beim Schreiben von [4] auch noch nicht bekannt). Diese "Sonderzeichen", zumindest aus amerikanischer Sicht sind sie etwas "Besonderes", werden jetzt, soweit ich das feststellen konnte, in allen GEMDOS-Funktionen korrekt verarbeitet. Ein Backspace über die Ausgabeposition Null hinaus wird immer noch nicht abgefangen.

GEMDOS-Funktionen

Nun werden die für den Programmierer relevanten Änderungen bzw. noch vorhandenen Fehler der GEMDOS-Funktionen 'C...', aufgezählt.

Bei den auf Dateien umgeleiteten Funktionen gilt allgemein, daß einige grobe Fehler beseitigt wurden, aber die Behandlung von Fehlern und des Dateiendes immer noch lückenhaft und inkonsistent ist. Einzig 'Cconws' macht eine Fehlerbehandlung, wie ich sie mir vorstelle. Abb. 6 ist ein "Update" der rechten Seite der Tab. 1 aus [5], die die Funktionsfähigkeit der I/O-Umleitung beschreibt. Die linke Seite liefere ich in einer späteren Folge nach.

Cconin, Cnecin

Device:

Die schon früher vorgesehene Erkennung der CTRL-Codes funktioniert nun.

Datei:

Das Dateiende wird immer noch nicht erkannt; es wird dann ein undefiniertes Zeichen zurückgegeben.

Cauxin, Crawcin

Device:

Keine direkten Änderungen. Insbesondere berücksichtigt 'Cauxin' immer noch nicht den GEMDOS-Eingabepuffer, der nach einer I/O-Umleitung Zeichen enthalten kann (s. [4]).

Datei:

Das Dateiende wird immer noch nicht erkannt; es wird dann ein undefiniertes Zeichen zurückgegeben.

Crawio

Device und Eingabevariante (Parameter gleich $00FF):

Wenn die Standardeingabe (Kanal 0) auf Drucker (Device 0) umgeleitet wurde, erhält man stets 0L (kein Zeichen verfügbar) zurück. Bisher gab es einen Absturz.

Datei und Eingabevariante:

Hier erhält man nun 0L (kein Zeichen verfügbar) zurück, wenn das 'Fread' fehlschlägt (i.allg. also das Dateiende erreicht ist). Unverständlich, daß dies nicht auch bei den anderen Eingabefunktionen gemacht wird, da ein Null-Zeichen immer noch besser ist also ein undefiniertes.

Datei und Ausgabevariante:

Es funktioniert nun überhaupt. Als Rückgabewert erhält man immer 0L, unabhängig davon, ob das 'Fwrite' erfolgreich war oder nicht.

Cconrs

Device:

Zeichen mit ASCII-Codes ab 128 werden nun nicht mehr fälschlicherweise als Steuerzeichen interpretiert und können somit

bei der Eingabe ohne Probleme benutzt werden. Die CTRL-Codes werden nun alle korrekt erkannt. Bei einem Abbruch mit 'CTRL-C' wird nun auch hier - wie bei allen anderen Funktionen, die 'CTRL-C' abfragen - der GEMDOS-Eingabepuffer gelöscht.

Datei:
Keine Änderungen. Die ATARI-Dokumentation behauptet, das Echo der Eingabe auf die Standardausgabe (Kanal 1) würde unterbleiben. Das ist jedoch nicht der Fall.

Cconis, Cauxis

Device:
Wenn der entsprechende Kanal auf Drukker (Device 0) umgeleitet wurde, erhält man stets 0L (kein Zeichen verfügbar) zurück.

Datei:
Das Dateiende wird korrekt abgefragt (wow!). Es wird -1L zurückgegeben, wenn das Dateiende nicht erreicht ist. Das ist der Fall, wenn *'fd_fpos'* und *'fd_len'* aus dem Filedescriptor (FD) ungleich sind. Der Wert 0L kann außer dem Dateiende auch signalisieren, daß der FD für die Datei gar nicht mehr existiert (z.B. nach einem Mediumwechsel).

Cconout

Device:
Das High-Byte des auszugebenden Zeichens, das sowieso Null sein sollte, wird nun explizit gelöscht und somit auch nicht an das BIOS weitergegeben.

Datei:
Es funktioniert nun überhaupt. Als Rückgabewert erhält man immer 0L, unabhängig davon, ob das *'Fwrite'* erfolgreich war oder nicht.

Cconws

Device:
Durch die Verwendung von *'register'*-Variablen ist diese Funktion etwas schneller geworden. Allerdings ist der "Overhead" immer noch erheblich.

Datei:
Die Ausgabe ist wesentlich schneller, da nur noch ein *'Fwrite'* gemacht wird. Hier werden sogar alle Fehler abgefangen. Ein eventueller GEMDOS-Fehler-Code von *'Fwrite'* wird zurückgegeben. Falls *'Fwrite'* nicht alle Zeichen schreiben konnte (z.B. bei vollem Speichermedium), erhält man den Fehler-Code ERROR (-1L), ansonsten die OK-Meldung *E_OK (0L)*.

Cprnout

Device:
Falls das Device der Drucker ist, erhält man wie oben diskutiert 0L bei erfolgreicher Ausgabe des Zeichens. Bei anderen Devices ist der Rückgabewert weiterhin undefiniert.

Datei:
Es funktioniert nun überhaupt. Als Rückgabewert erhält man immer 0L, unabhängig davon, ob das 'Fwrite' erfolgreich war oder nicht.

Cauxout

Device:
Keine Änderungen.

Datei:
Es funktioniert nun überhaupt. Als Rückgabewert erhält man immer 0L, unabhängig davon, ob das *'Fwrite'* erfolgreich war oder nicht.

Cconos, Cauxos, Cprnos

Device:
Keine Änderungen.

Datei:
Es wird nun korrekt -1L (statt 0xff) geliefert, um anzuzeigen, daß jederzeit Zeichen angenommen werden können.

Vorschau

So, das war's für heute. Das nächste Mal möchte ich Ihnen dann etwas über die Dateiverwaltung erzählen. Vielleicht ist das dann schon der letzte Teil dieser Miniserie.

Alex Esser

Literatur:

[1] A.Esser: *"TOS intern", ST-Computer Sonderheft Nr. 2, 1987*

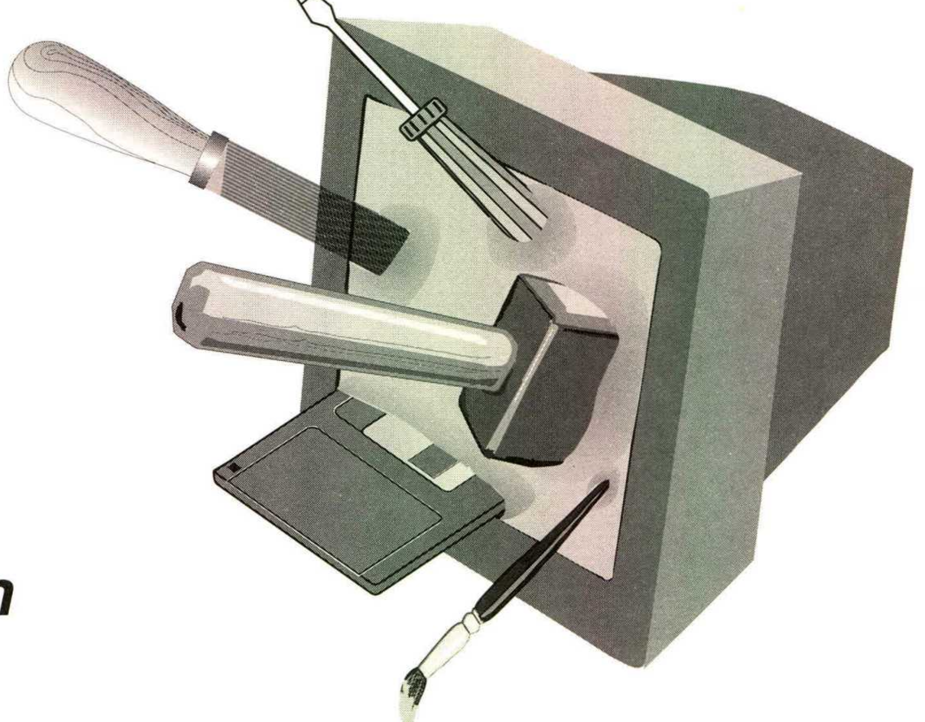
[2] A.Esser: *"Auf der Schwelle zum Licht - Uhrzeit und Datum", ST-Computer 2/89*

[3] A.Esser: *"Auf der Schwelle zum Licht - Teil 1", ST-Computer 12/87*

[4] A.Esser: *"Auf der Schwelle zum Licht - Zeichenorientierte Geräte", ST-Computer 10/88*

[5] A.Esser: *"Auf der Schwelle zum Licht - I/O-Umleitung", ST-Computer 12/88*

Programmer's Toolbox Dateien



Teil 1: Einleitung und Grundlagen

Eigentlich hat man sich ja daran gewöhnt, an die Maus, die Fenster und an alles, was sonst noch zu einer grafischen Benutzeroberfläche wie der des ATARI ST gehört. Trotzdem, so manchmal juckt es mich doch in den Fingern, das Hacken, das Malträtieren von Tastaturen durch die manuelle Eingabe von Kommandozeilen.

Verstärkt wird dieses Jucken, wenn sich beispielsweise folgende Aufgabenstellung einstellt:

Sichere alle C-Sourcen aus Verzeichnis C:\XYZ\ auf Diskette!

Da hat man es nämlich wieder: Durch jahrelangen mehr oder weniger ungebrochenen Fortschritt in der Entwicklung der Mensch-Maschine-Kommunikation ist man nun gezwungen, seine Brille aus der Ecke hervorzukramen (der Autor ist ziemlich kurzsichtig), und das lästige Suchen von Dateien mit der Endung "C" beginnt. Hat man endlich fast alle derartigen Dateien beisammen, dann klickt die dumme Maus doch wieder an der falschen Stelle, und die Suche beginnt erneut. Der Rest ist Legende. Dabei könnte es doch so einfach sein:

```
CP C:\XYZ\*.C A:\
```

Probleme der oben geschilderten Art haben mich auf die Idee zu dieser Artikelserie gebracht. Aus der Idee wurde Handlung und aus der Handlung Software: die Software, die Sie in den nächsten Folgen der ST Computer innerhalb dieser Serie abgedruckt finden. Es handelt sich dabei um eine textuelle Benutzeroberfläche. Damit ich im folgenden aber nicht immer "textuelle Benutzeroberfläche" sagen muß, nenne ich sie ab jetzt bei ihrem englischen Namen und der lautet "Shell". Die Shell besitzt einen kleinen Kern von eingebauten Kommandos und eine weitaus größere Anzahl von externen (Kommandos die für sich genommen eigenständige Programme darstellen). Diese Konzeption bietet eine Reihe von Vorteilen: Der erste Vorteil besteht darin, daß die Shell klein ist. Nachgeladen werden immer nur die gerade benötigten Kommandos. Weiterhin besteht die Möglichkeit, die Shell durch Hinzufügen von eigenen externen Kommandos beliebig zu erweitern, ohne sie im Kern ändern zu müssen. Außerdem ist es möglich, die externen Shell-Kommandos außerhalb der Shell anzuwenden. Deren Einführung und Programmierung erfolgt dabei in insgesamt drei thematischen Blöcken:

1. Rund ums Dateisystem

Der erste Block umfaßt die Shell sowie einige Kommandos, die zur Dateiverwaltung benötigt werden (Ansehen von Verzeichnissen, Verschieben, Löschen und Kopieren von Dateien etc.). Dabei werden grundlegende Begriffe rund um das Dateisystem erklärt, und es erfolgt eine Vorstellung der für das Thema wesentlichen Bestandteile des Betriebssystems. Weiterhin wird gezeigt, wie Pipelining, Ein/Ausgabe-Umlenkung und Kommandoprozeduren mit und innerhalb der Shell realisiert werden können. Insgesamt werden zu dieser Thematik acht Folgen erscheinen.

2. Text

Der zweite thematische Block behandelt eine spezielle Dateiart: Text. Es werden Kommandos zum Ansehen, Sortieren und Durchsuchen von Text eingeführt. Für dieses Thema sind vier Folgen vorgesehen.

3. Vom Packen und Verschlüsseln

Im dritten Block geht es dann um das Packen und Verschlüsseln von Dateien. Es wird gezeigt, wie man Dateien platzsparend ablegt. Außerdem werden unterschiedliche Verschlüsselungsarten einge-

führt, untersucht und programmiert. Die Anwendung ist hier vielfältig und reicht vom Paßwortschutz für bestimmte Programme bis hin zur (fast) sicheren Verschlüsselung ganzer Dateien. Der dritte thematische Block wird voraussichtlich ebenfalls vier Folgen umfassen.

Die Kommandos innerhalb aller Artikel sind dabei angelehnt an entsprechende Kommandos unter dem Betriebssystem UNIX. Ihre Programmierung ist in C erfolgt. Verwendet wurde der Laser C Compiler. Da jedoch keine speziellen Aspekte dieses Programms benutzt werden, dürften die Kommandos auch mit einem beliebigen anderen C-Compiler für den ATARI ST compilierbar sein. Voraussetzung dabei ist eine Betriebssystemschnittstelle (GEMDOS) sowie das Vorhandensein der UNIX-kompatiblen Funktion der Sprache C.

An dieser einleitenden Stelle möchte ich nun noch einige Worte zum Titel der Serie sagen: Man kann sicherlich sagen, daß die in den einzelnen Kapiteln vorgestellten Kommandos so etwas wie "Programmwerkzeuge" darstellen. Eine Ansammlung davon könnte man dann wohl auch als "Programmierers Werkzeugkasten" bezeichnen. Das war meine Ausgangsidee. Etwas "eingeenglischt" hört es sich nicht mehr ganz so bieder an, nämlich gerade "Programmer's Toolbox". Über die Verwendung von englischen Begriffen anstelle der deutschen kann man sicherlich streiten. Auch ich verwende im folgenden ein "gesundes Mischmasch" beider Sprachen. Mal ehrlich: "Cowboy" ziehen Sie doch wohl auch der Bezeichnung "Kuhhirte" vor. Mir geht es genauso. Der Rest des Titels - "Dateien" - weist dann noch darauf hin, daß ausschließlich Programmwerkzeuge - "Tools" - behandelt werden, die sich mit der Bearbeitung von Dateien beschäftigen.

Rund ums Dateisystem

Kommen wir nun zur Inhaltsangabe des ersten thematischen Blocks: Heute erfolgt eine Einführung grundlegender Begriffe wie Gerät, Pfad und Kommando. Ihre Bedeutung ist nicht auf den ATARI ST begrenzt, vielmehr finden sich die hier vorgestellten Strukturen auch in anderen Betriebssystemen. Als Beispiel sei hier nur auf MS-DOS und UNIX verwiesen. In der nächsten Folge geht es dann ins Innere des ST. Es werden die GEMDOS-Betriebssystemfunktionen des ATARI ST vorgestellt. Derartige Textstellen haben immer so etwas wie Nachschlagecharakter. Deshalb werde ich mich nicht auf die von mir benötigten Funktionen beschränken. Stattdessen finden alle Funktionen des GEMDOS kurz Behandlung. Nachfolgend beginnt die eigentliche Programmierarbeit. Zunächst werden einige Hilfsfunktionen programmiert, die so sinnvolle Aufgaben wie das Setzen der Systemzeit oder das Kopieren, das Bewegen und das Löschen von Dateien bewerkstelligen (3. Teil). Mit diesen Hilfsfunktionen ist es dann möglich, einige Kommandos zu programmieren. Hierzu zählen die folgenden (4. Teil):

CD	- Ändern des Arbeitsverzeichnisses
PWD	- Ausgabe des Arbeitsverzeichnisses
DATE	- Auslesen/Setzen des Systemdatums
TIME	- Auslesen/Setzen der Systemzeit
DF	- Auslastungsstatistik für Dateisysteme

Danach werden wir uns mit der Expansion von Argumenten beschäftigen (5. Teil) und Kommandos implementieren, die Argumentexpansion benutzen (6. und 7. Teil). Dies sind die folgenden Kommandos:

LS	- Anzeigen von Dateiinformationen
CP	- Dateien kopieren
MV	- Dateien bewegen
RM	- Löschen von Dateien
RMDIR	- Löschen von Verzeichnissen
MKDIR	- Erzeugen von Verzeichnissen
CHMOD	- Ändern des Dateizugriffsmodus'
TOUCH	- Aktualisierung der Modifikationszeit von Dateien

Zum Abschluß des ersten Artikelblocks wird die Shell (Kommando SSH) programmiert (8. Teil). Es wird gezeigt, wie man korrekte Eingaben erkennt und wie Pipelining und Kommandoprozeduren realisiert werden.

Von Geräten, Pfaden und Verzeichnissen

Doch damit genug vorausgeschaut, beginnen wir mit der Begriffsklärung und hier mit dem Begriff Gerät. In einer abstrakten Betrachtungsweise stellt sich ein Gerät als Eingabe- oder Ausgabemöglichkeit für einen Computer dar. Reine Eingabegeräte sind etwa die Tastatur und die Maus. Ausgabegeräte sind Bildschirm und Drucker. Daneben existieren eine Reihe von Geräten, die Eingabe- und Ausgabegeräte sein können, etwa die Floppy- oder die Hard-Disk. Normalerweise werden diese Geräte genutzt, um große, zeitweise nicht benötigte Datenmengen zwischenzuspeichern und für spätere Anwendungen zur Verfügung zu halten. Die Ablage der Daten erfolgt dabei in Dateien. Im Gegensatz zu den reinen Eingabe- bzw. Ausgabegeräten benötigen Geräte mit Eingabe- und Ausgabefähigkeiten einen Speicher, der mitunter ein Vielfaches der Kapazität des Hauptspeichers besitzt und damit eine große Anzahl von Dateien aufnehmen kann. Unter diesem Umstand ist es verständlich, daß eine bestimmte Organisationsform für die Dateien gefunden werden muß, um einen gewissen Grad von Übersichtlichkeit zu erreichen. Der Begriff Dateisystem ist dabei nichts weiter als eine andere Bezeichnung für "Organisationsform für Dateien". Das von ATARI für den ST benutzte Dateisystem lehnt sich dabei an eine Organisationsform an, die bereits etwas betagt ist, nichtsdestotrotz aber bisher recht erfolgreich war: die XENIX-Struktur. Innerhalb von Betriebssystemen, die gemäß der XENIX-Struktur organisiert sind (UNIX, MS-DOS, TOS...), erfolgt die Strukturierung von Dateien mit Hilfe von Verzeichnissen. Ein Gerät enthält hierbei genau ein sogenanntes Hauptverzeichnis. Das Hauptverzeichnis erlaubt den Eintrag von Dateien und Unterverzeichnissen. Auch Unterverzeichnisse können ihrerseits wiederum Verzeichnisse enthalten und so fort. Es ergibt sich die Möglichkeit einer hierarchischen Schachtelung von Verzeichnissen, womit wir bei Abb.1.1 sind. Will man nämlich derartige hierarchische Schachtelungen in ihrer Gesamtheit darstellen, bietet sich die dort gewählte baumartige Darstellungsweise an. Die Wurzel des Baumes ist dabei das Hauptverzeichnis und wird mit dem Zeichen '\' benannt. Unterverzeichnis- und Dateinamen sind mit gewissen Einschränkungen (maximal acht Buchstaben für den Dateinamen; ein Punkt; maximal drei Buchstaben für den Dateityp) frei wählbar und bilden die "Knoten" und die "Blätter" des Baumes. Die "Äste" des Baumes stellen je nach Betrachtungsweise eine der beiden Relationen "ist Unterverzeichnis von" bzw. "ist Unterverzeichnis in" dar. Ein Pfad bis zu einer Datei oder einem Verzeichnis ist jeweils der Weg vom Hauptverzeichnis bis zu der gewünschten Datei bzw. dem Verzeichnis. Zur eindeutigen Identifizierung einer Datei innerhalb eines Dateisystems wird in Systemen mit XENIX-Struktur der Pfadname der Datei verwendet. Der Pfadname einer Datei ist die Aneinanderreihung aller Unterverzeichnisnamen, beginnend vom Hauptverzeichnis bis hin zur Datei. Die Unterverzeichnisnamen werden beim ATARI ST dabei mit dem Zeichen '\' voneinander getrennt. Für die Datei *KAP_1.DOC* innerhalb von Abb.1.1 gilt damit der Pfadname:

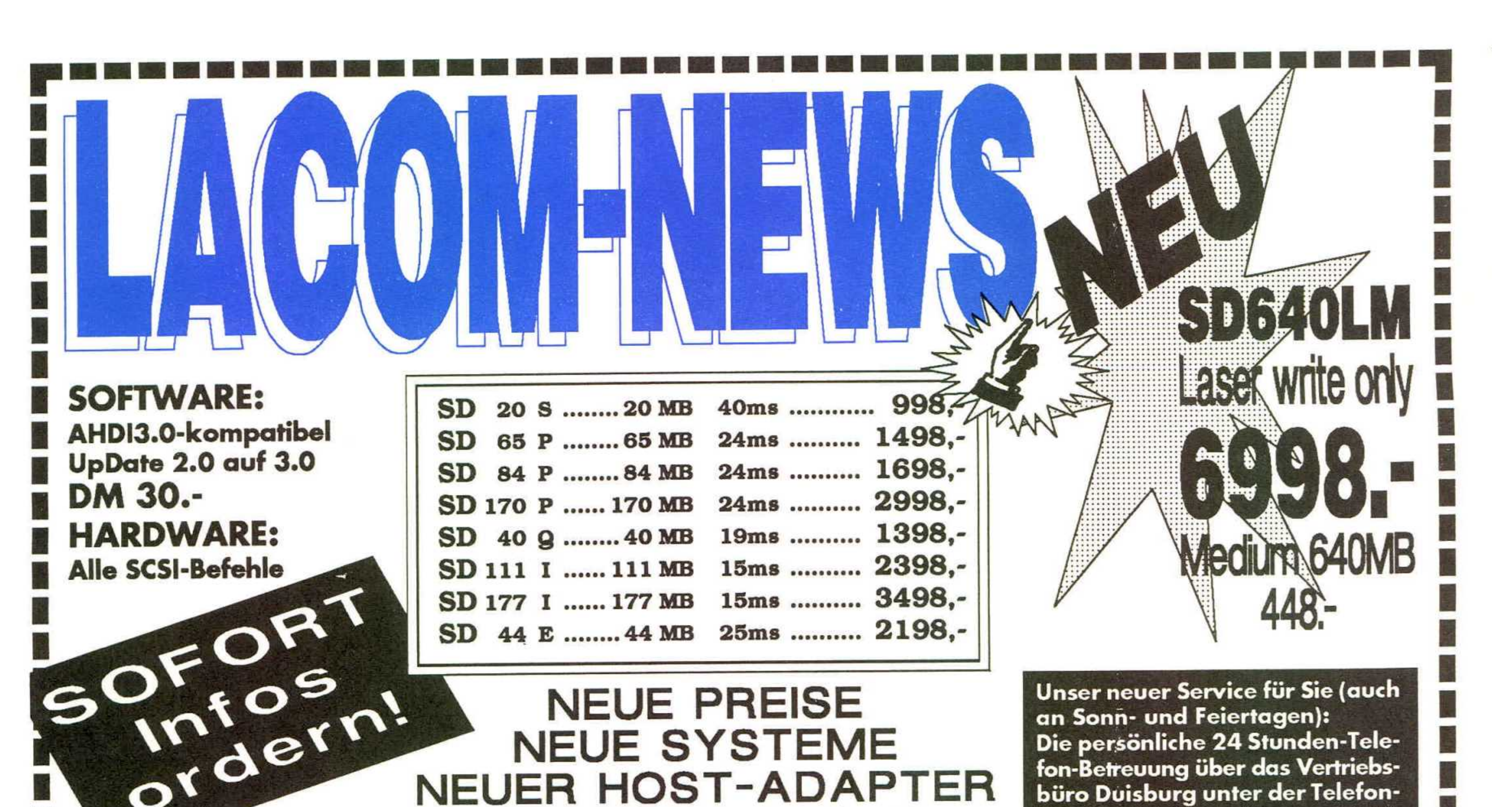
COMPUTERWARE BRINGT SCHWUNG IN IHREN ATARI



Hard Disk Sentry: Datenverlust und Fehler in den Verzeichnissen - ein echter Alptraum! Mit dem Sentry kann Ihnen das nicht passieren. Dieses Programm beugt vor, indem es die Verzeichnisse prüft und in der Lage ist, mögliche Fehler sofort zu reparieren. Sie können übrigens auch die Zugriffszeiten zu Ihren Dateien verkürzen, denn der Sentry „räumt auf" - das ist das Stück Sicherheit mehr, das Sie bald schon nicht mehr vermissen möchten! Unverbindliche Preisempfehlung: 139,- DM. Überzeugen Sie sich bei Ihrem Atari-Fachhändler. Von ihm bekommen Sie auch Prospekte.

COMPUTERWARE

Computerware • Gerd Sender • Weißer Straße 76 • D-5000 Köln 50 • Telefon: 0221-392583 • Telefax: 0221-396186
Schweiz: DataTrade AG Zürich • Telefon: 01-2428088 • **Österreich:** Reinhard Temmel GmbH • Telefon: 06244-70810



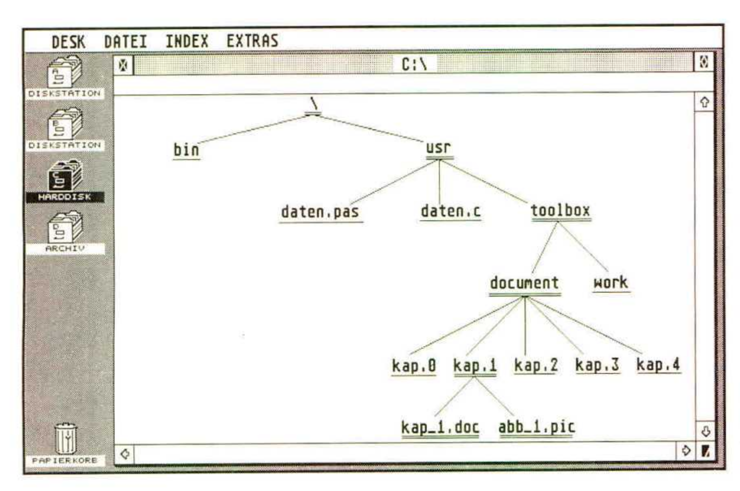


Bild 1: Ein Inhaltsverzeichnis einmal anders

\USR\TOOLBOX\DOCUMENT
\KAP.1\KAP_1.DOC

Da diese Pfadnamen bei stark geschachtelten Dateisystemen mitunter recht lang werden, führt man den Begriff des Arbeitsverzeichnisses bzw. des aktuellen Verzeichnisses ein. Das Arbeitsverzeichnis ist unter allen Verzeichnissen des Dateisystems frei wählbar. Nach Wahl eines Arbeitsverzeichnisses können Zugriffe auf Dateien oder weitere Verzeichnisse relativ zum Arbeitsverzeichnis erfolgen. Relative Pfadnamen unterscheiden sich äußerlich kaum von gewöhnlichen (absoluten) Pfadnamen. Sie erkennt man lediglich am Fehlen der führenden Wurzelkennung '\'. Allerdings sind relative Pfade meistens deutlich kürzer, da bei ihnen nur der Pfad ab dem Arbeitsverzeichnis angegeben werden muß. Mit dem Arbeitsverzeichnis *\USR\TOOLBOX\DOCUMENT* könnte so etwa die Datei *KAP_1.DOC* in Abb.1.1 mit folgendem Pfad selektiert werden:

KAP.1\KAP_1.DOC

Auch Dateien außerhalb des Arbeitsverzeichnisses lassen sich relativ zum Arbeitsverzeichnis beschreiben. Will man etwa aus dem Arbeitsverzeichnis *\USR\DATEN.C* die Datei *KAP_1.DOC* selektieren, so geschieht dies mit folgendem Pfad:

..\TOOLBOX\DOCUMENT\KAP.1\KAP_1.DOC

Die Verwendung der beiden Punkte innerhalb der Pfadnamen bedeutet dabei soviel wie: "Gehe ein Verzeichnis im Dateisystem zurück!". Tatsächlich ist ".." ein Verzeichnis, das in jedem Unterverzeichnis "unsichtbar" vorhanden ist und das auf das jeweilige Vorgängerverzeichnis weist. Damit ist es auch an den äußeren Enden eines Dateisystems bequem möglich, Dateien in unmittelbarer Nähe anzusprechen, ohne den absoluten Pfadnamen anzugeben. Natürlich kann man ".." auch in absoluten Pfaden verwenden, besonders sinnvoll ist das aber nicht:

\USR\..\USR\TOOLBOX\..\TOOLBOX\
DOCUMENT\KAP.1\KAP_1.DOC

Ich kann mir nicht helfen, irgendwie sieht das sehr nach Stottern aus.

Von Kommandos, Optionen und Argumenten

Weitere Grundbegriffe gruppieren sich um den Begriff des Kommandos:

Unter einem Kommando versteht man eine Handlungsanweisung für den Rechner. Damit ist *Kommando* an sich nichts weiter als eine andere Bezeichnung für den allgemeineren und geläufigeren Begriff des *Programms*. In Abgrenzung zu allgemeinen Programmen wird der Begriff des Kommandos im folgenden jedoch mit einer eingeschränkten Bedeutung benutzt: Als Kommandos sollen nur diejenigen Programme bezeichnet werden, die auf die grafischen Möglichkeiten des ST verzichten. Dies sind die sogenannten TOS-Programme. Im ST-Betriebssystem enden sie entweder auf *TOS* oder auf *TTP*. Interessanter sind dabei die auf *TTP* endenden Programme, da sie dem Anwender die Möglichkeit geben, dem Programm durch eine Kommandozeile Anweisungen zu erteilen (TTP=TOS Takes Parameter). Werden TTP-Programme aus dem Desktop gestartet, wird zur Eingabe der Kommandozeile ein Dialog durchgeführt. Diese Kommandozeile wird beim direkten Aufruf von TTP-Programmen (also unter Umgehung des Desktop) dem Betriebssystemaufruf *Pexec* als (String-)Parameter mitgegeben.

Innerhalb dieser Artikelserie soll nun eine dritte Möglichkeit realisiert werden: die Eingabe von Kommandos als Text innerhalb einer Shell. Einen ersten Eindruck, wie dies auszusehen hat, möchte ich mit Abb.1.2 und dem zunächst noch hypothetischen Kommando *LS* vermitteln. Hier dargestellt ist eine Kommandozeile, wie sie in der späteren Implementierung von unserer Shell verstanden werden soll. Die einzelnen Parameter der Kommandozeile werden dabei durch Leerzeichen vom Kommandonamen und voneinander getrennt. Weiterhin werden zwei Arten von Parametern unterschieden: Optionen und Argumente. Ein Parameter wird als Option bezeichnet, wenn er einem Kommando Auskunft darüber gibt, **wie** es zu arbeiten hat. Dagegen gilt ein Parameter als Argument, wenn er dem Kommando angibt, **was** zu bearbeiten ist. Üblicherweise sind Optionen mit einem führenden '-' gekennzeichnet. Zusätzlich gilt, daß alle Optionen eines Kommandos vor dem Auftreten des ersten Argumentes angegeben sein müssen, ansonsten werden sie nicht mehr als Optionen erkannt, sondern als Argumente behandelt.

Anmerkung: Bitte beachten Sie, daß die Differenzierung zwischen Optionen und Argumenten nur auf einer Konvention beruht. Der nachfolgend beschriebene Übergabemechanismus behandelt beide Parameterarten gleich.

Mit der Einschränkung von Kommandos auf Text als einziges Mittel zur Ein- und Ausgabe vereinfachen sich die Wechselwirkungen zwischen Kommandos und ihrer Umgebung drastisch. Lediglich Rückfragen zur Anforderung von Steuerinformationen und Ergebnisausgaben sind neben der eigentlichen Wirkung von Kommandos möglich. Der Datenkanal, aus dem ein Kommando Steuerinformationen liest, heißt dabei Standardeingabekanal. Umgekehrt heißt der Datenkanal, über den Steuerinformationen angefordert werden und der die Ergebnisausgabe ermöglicht, Standardausgabekanal. Für beide Kanäle ist die Konsole voreingestellt, d.h. Ein- und Ausgaben geschehen

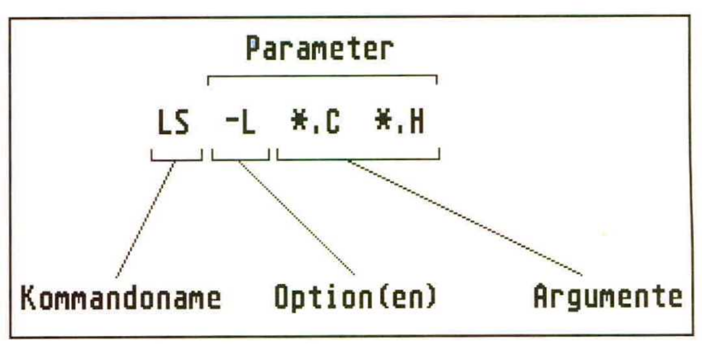


Bild 2: Beispiel für ein Kommando

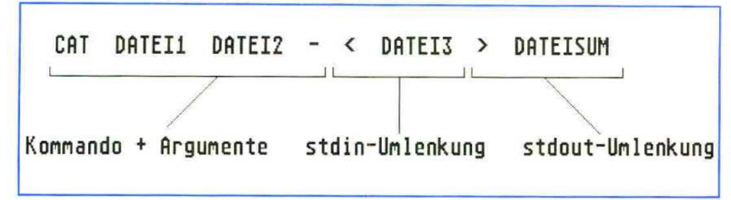


Bild 3: Beispiel für Umlenkungen der Ein- und Ausgabe

interaktiv mit Bildschirm und Tastatur. Zur Flexibilisierung von Programmabläufen sind jedoch Umlenkungen beider Kanäle erlaubt, um z.B. den wiederholten Ablauf eines Kommandos mit gleichen Steuerinformationen zu erlauben. Abb.1.3 zeigt dazu ein Beispiel. Zu sehen ist, daß die Umlenkung der Standardeingabe mit einem '<'-Zeichen vorgenommen wird. Entsprechendes gilt für die Standardausgabe und das '>'-Zeichen. Das Beispiel würde etwa die Standardeingabe des (auch noch hypothetischen) Kommandos *CAT* aus der Datei *DATEI3* entnehmen. Die Standardausgabe erfolgt in die Datei *DATEISUM*.

Anmerkung: Das '-' ist in Abb.1.3 keine Option, sondern kennzeichnet die Stelle, an der die Standardeingabe in die Ausgabe des Kommandos *CAT* übernommen wird. *CAT* wird erst im zweiten Block der "Programmer's Toolbox" realisiert.

Beide Umlenkungsarten werden direkt durch TOS unterstützt, d.h. wenn beim Aufruf eines TTP-Programms (aus dem Desktop oder mit *Pexec*) eine Kommandozeile mit Umlenkungen angegeben wird, wird das Programm mit den entsprechenden Werten für die Standardkanäle durchgeführt.

Soviel zunächst zu der Sicht aus dem Betriebssystem auf Kommandos und ihre Parameter. Bevor nun gezeigt wird, wie die übergebenen Parameter dem Kommando zugänglich gemacht werden, möchte ich kurz Listing 1.1 vorstellen. Es zeigt nicht viel: Einige wenige Konventionen, die ich im weiteren Ablauf verwenden möchte, werden hier in Form der Header-Datei *LOCAL.H* eingeführt.

Das erste "echte" Listing bezieht sich aber bereits auf das Thema: Listing 1.2 zeigt eine "abgespeckte" Version des eigentlich erst im zweiten Block vorkommenden Kommandos *ECHO*. Diese Version erhält die Bezeichnung *ECHOSIMP* und macht nichts anderes, als die ihm übergebenen Argumente auf den Standardausgabekanal zu schreiben. Abschließend wird das ganze dann noch mit einem Zeilenvorschub versehen. Die einzige erlaubte Option besteht dabei darin, den Zeilenvorschub abzustellen (Option "-N"). In einer gebräuchlichen Schreibweise läßt sich die Parametrisierung von *ECHOSIMP* folgendermaßen beschreiben:

ECHOSIMP [-N] [Argument...]

Die eckigen Klammern stehen dabei für optionale Komponenten. Argument ist ein Platzhalter für ein beliebiges Argument und die Induktionspunkte besagen, daß eine beliebige Anzahl von Argumenten erlaubt ist. Insbesondere ist es damit erlaubt, *ECHOSIMP* ohne Argumente aufzurufen.

Der Zweck dieser vorgezogenen Variante des Kommandos *ECHO* ist die Demonstration des Parameterübergabemechanismus' anhand eines einfachen Beispiels. Wenn wir uns nun der Implementierung von *ECHOSIMP* zuwenden, ist zunächst die Funktion main von Interesse. Die beim Programmaufruf übergebenen Parameter tauchen hier nämlich als Funktionsparameter der Hauptfunktion main wieder auf. Der erste Funktionsparameter von *main* *(argc)* gibt dabei an, wieviele Parameter insgesamt übergeben werden. Der zweite Parameter *(argv)* weist auf ein Feld von Strings, das die aktuellen Werte der Parameter enthält. Dabei ist zu beachten, daß in der Laser C-Implementierung des Parameterübergabemechanismus' der erste String immer leer bleibt, nichtsdestotrotz in der Berechnung für *argc* aber mitgezählt wird. Betrachten wir einen *ECHOSIMP*-Aufruf der folgenden Form:

ECHOSIMP -N Hallo Welt

Es ergeben sich folgende Zustände für die Parameter von *main*:

argc : 4
argv[0] : ""
argv[1] : "-N"
argv[2] : "Hallo"
argv[3] : "Welt"

Die das Kommando realisierende Funktion *echosimp* zeigt nun, wie die Parameter interpretiert werden. Zunächst wird hier die Option abgefragt und gegebenenfalls ein entsprechender Merker gesetzt (*newline*), dann erfolgt die Ausgabe der übrigen Parameter als Argumente. Zum Abschluß wird noch, in Abhängigkeit vom Wert des Merkers, ein Zeilenvorschub vorgenommen.

An dieser Stelle möchte ich auf eine Eigenheiten des ST-Betriebssystems hinweisen, da sie auch Eingang in die Implementierung meiner Programme findet. Im Gegensatz beispielsweise zu UNIX gehört TOS zu den Betriebssystemen, die nicht zwischen Klein- und Großschreibung differenzieren. In der Praxis sieht dies so aus, daß alle Betriebssystemaufrufe ausschließlich mit Großbuchstaben arbeiten. Entsprechend führen sie vor ihrer Ausführung zunächst eine Konvertierung aller Klein- in Großbuchstaben durch. Ausnahmen bestätigen die Regel, und eine der Ausnahmen bei der String-Konvertierung ist gerade die Funktion *Pexec*. Im Gegensatz zu anderen Betriebssystemaufrufen wird die *Pexec* übergebene Kommandozeile nicht konvertiert. Die Konvertierung erfolgt hier bereits zuvor, nämlich zwischen der Ausführung des TTP-Dialoges im Desktop und dem Aufruf von *Pexec*. Wird *Pexec* nicht aus dem Desktop ausgeführt, erfolgt damit auch keine Konvertierung! Auswirkungen dieser Tatsache bestehen z.B. darin, daß Kommandos eigenständig dafür zu sorgen haben, daß Kleinbuchstaben in Parametern richtig interpretiert werden. Auf der anderen Seite ist ein Kommando wie *ECHO* (oder *ECHOSIMP*) nicht darauf eingeschränkt, ausschließlich großgeschriebene Texte auszugeben.

Vorausschau

Ja, das war's dann auch schon für heute. In der nächsten Folge der "Programmer's Toolbox" folgt die Einführung der GEMDOS-Betriebssystemfunktionen. Erst hatte ich vor, nur die Funktionen herauszupicken, die innerhalb der Serie benötigt werden. Da dann aber doch die meisten Funktionen betroffen gewesen wären, denke ich mir, daß es nicht schaden kann, ein komplettes Kurz-Reference des GEMDOS anzugeben. Wie Sie feststellen werden, ist das nicht so umfangreich, wie man denkt. In diesem Sinne - tschüß bis zum nächsten Mal.

Dirk Brockhaus

```
 1:  /*
 2:   * Listing 1.1, Datei : local.h
 3:   * Modifikationsdatum : 17-Nov-89
 4:   * Abhängigkeiten     : -
 5:   */
 6:
 7:  typedef short        BOOLEAN;
 8:
 9:  #define FALSE        0
10:  #define TRUE         1
11:
12:  #define ERROR        (-1L)
13:
14:  #define todigit(ch)  ((ch) - '0')
15:
16:  #define isoctal(ch)  ((ch) >= '0'
                           && (ch) <= '7')
```

```
 1:  /*
 2:   * Listing 1.2, Datei : echosimp.c
 3:   * Programm           : ECHOSIMP - Ausgabe der
 4:   *                      Argumente auf der Stan-
                             dardausgabe (einf.Version)
 5:   *                      (C) MAXON Computer GmbH
 6:   * Modifikationsdatum : 17-Nov-89
 7:   * Abhängigkeiten     : stdio.h, string.h, local.h
 8:   */
 9:
10:  #include <stdio.h>
11:  #include <string.h>
12:  #include "local.h"
13:
14:  /*
15:   * Funktion      : echosimp
16:   *
17:   * Parameter     : echosimp(argc, argv);
18:   *                 short argc;
19:   *                 char  *argv[];
20:   *
21:   * Aufgabe       :
22:   *
23:   * Interpretierung der durch <argc> und <argv>
24:   * spezifizierten Parameterliste gemÑû den Fest-
25:   * legungen des Kommandos echosimp.
26:   */
27:
28:  void echosimp(argc, argv)
29:  short argc;
30:  char  *argv[];
31:  {    short   i,
32:               start = 1;
33:       BOOLEAN newline = TRUE;
34:
35:       if (argc > 1) {
36:            if (strcmp(argv[1], "-N") == 0 ||
37:                strcmp(argv[1], "-n") == 0) {
38:                newline = FALSE;
39:                start = 2;
40:            }
41:            for (i = start; i < argc; i++)
42:                printf("%s", argv[i]);
43:       }
44:       if (newline)
45:            printf("\n");
46:  }
47:
48:  void main(argc, argv)
49:  short argc;
50:  char  *argv[];
51:  {    echosimp(argc, argv);
52:       exit(0);
53:  }
```

KOPFNÜSSE

Fußzeilen in LaTex

TeX und das komfortable Makropaket LaTeX erfreuen sich auch auf dem ST immer größerer Beliebtheit - nicht zuletzt haben dazu die Public-Domain-Implementierungen von Christoph Strunk sowie Stefan Lindner und Lutz Birkhahn beigetragen. Wie erledigt der Benutzer ohne größere TeX-Programmierkenntnisse aber vermeintlich einfache Dinge wie das Ändern von Kopf- und Fußzeilen?

In einem Wortprozessor wie beispielsweise Wordplus gibt es meist eine sehr einfache Möglichkeit, in einer Dialogbox die Kopf- und Fußzeilen für das Dokument festzulegen. Über Sonderzeichen können dabei auch variable Bestandteile wie Seitennummer oder Ausdruckdatum verwendet werden

Wer mit LaTeX arbeitet, kennt auch eine Möglichkeit, das Aussehen einer Seite festzulegen: über das *\pagestyle*-Kommando. Dabei bietet das normale LaTeX-Format vier Einstellungen an: *plain*, *empty*, *headings* und *myheadings*. Ein Kommando *\pagestyle{plain}* im Eingabetext legt fest, daß die Kopfzeile leer ist und im Fuß die Seitennummer steht. Setzt man *headings* ein, bleibt der Fuß leer, dafür steht in der Kopfzeile je nach Dokumentenart (bei *\documentstyle* festgelegt) eine Kombination aus der Überschrift des laufenden Kapitels, dessen Nummer und der Seitenzahl. *empty* schließlich legt einen Seitenstil fest, bei dem keinerlei Informationen in Kopf- und Fußzeile erscheinen.
Will man das Layout nur für die momentan abgearbeitete Seite ändern, verwendet man anstelle von *\pagestyle* das Kommando *\thispagestyle* mit demselben Argument. Die Voreinstellung des Seitenstils ist übrigens *plain*, wenn man das originale LaTeX verwendet.

Eigene Kopfzeilen

Die Erläuterung der angebotenen Seitenstile war noch nicht komplett: Es fehlt *myheadings* in der Aufzählung. Und genau damit läßt sich ein eigenes Format der Kopfzeilen definieren, wozu die LaTeX-Befehle *\markright* und *\markboth* verwendet werden. Wir werden uns zunächst mit dieser Möglichkeit beschäftigen, können aber schon vermerken, daß offenbar die Fußzeilen so nicht zugänglich sind.

Ist der Seitenstil *myheadings* eingestellt, kann mit *\markright* der Text der Kopfzeile der jeweils rechten (also in einem doppelseitigen Dokument die ungeraden) Seiten eingestellt werden. Ein Kommando

```
\markright{Protokoll}
```

im Eingabetext führt zu einem Seitenkopf wie in Bild 1. Links steht der vorher festgelegte Text, rechts außen steht die Seitennummer. Die Darstellung der Seitennummer und die Schriftart werden übrigens nicht verändert.

Natürlich kann man auch veränderliche Bestandteile in den Kopf aufnehmen. *\jobname* liefert den Namen des gerade bearbeiteten Dokuments; *\today* setzt das aktuelle Datum. Mit

```
\markright{\jobname(\today)}
```

ergibt sich nun eine Kopfzeile wie bei Seite 3 im Bild 1. Man kann auch andere TeX-Befehle verwenden, beispielsweise die Änderungen der Schriftart. Um das gerade gezeigte Beispiel optisch etwas zu verbessern, fügt *\quad* einen Leerraum ein und für die Klammerung des Datums finden mathematische Sondersymbole Verwendung:

```
\markright{\jobname\quad
$\rhd$\today$\lhd$}
```

Das Ergebnis sehen Sie wiederum im Bild. Warum gibt es aber nun *\markright* und *\markboth* aber kein *\markleft*? LaTeX kennt Dokumente, die einseitig sind, und

solche, die, wie in einem Buch, linke und rechte Seiten haben. Letztere werden durch die Stiloption *twoside* gekennzeichnet und verwenden unterschiedliche Kopfzeilen. Die Seitennummer steht dabei immer außen, also auf linken Seiten links und auf rechten Seiten rechts. Daher gibt es also einen linken und einen rechten Kopf, die unterschiedlich aussehen. Beim Seitenstil *headings* wechseln auch die Texte in der Kopfzeile: Links steht die Kapitelüberschrift, rechts die des Abschnitts (bei den Dokumentenstilen *book* und *report*).

Bei einem einseitigen Dokument wird immer nur der rechte Kopf verwendet, daher reicht das Kommando *\markright*. Bei doppelseitigen Schriftstücken sind zwei Kopfdefinitionen nötig, daher hat *\markboth* zwei Argumente, wobei das erste die linke Kopfzeile beschreibt. *\markboth* kann natürlich auch bei einseitigen Dokumenten verwendet werden, der linke Kopf wird eben nicht verwendet.

Eine völlig freie Einstellung der Kopfzeilen ist so aber noch nicht zu machen. Zwar läßt sich der Text einstellen, festgelegt sind aber noch die Seitenzahl, die Ausrichtung von Text und Seitenzahl sowie das eigentliche Layout der Kopfzeile. Leslie Lamport verwendet in seinem LaTeX-Buch [1] Kopfzeilen mit halbfetter Seitennummer, die zudem unterstrichen sind. Um das zu machen, muß tiefer in LaTeX eingestiegen werden. Dazu sollten Sie die Kapitel 11, 12 und 20 aus dem TeXbook kennen [2] oder Kapitel 7 und 8 aus dem Buch von Norbert Schwarz [3].

```
% from book.sty
\def\ps@headings{\let\@mkboth\markboth
  \def\@oddfoot{}
  \def\@evenfoot{}
  \def\@evenhead{\rm \thepage\hfil \sl\leftmark}
  \def\@oddhead{\hbox{}\sl \rightmark \hfil\rm\thepage}
  \def\chaptermark##1{\markboth {\uppercase{\ifnum
\c@secnumdepth>\m@ne
    \@chapapp\ \thechapter. \ \fi ##1}}{}}
  \def\sectionmark##1{\markright{\uppercase{\ifnum \c@secnumdepth
>\z@
    \thesection. \ \fi ##1}}}}
```

Listing 1

Protokoll	2
heads(30. Januar 1990)	3
heads ▷30. Januar 1990◁	4

Bild 1

```
\makeatletter
\def\ps@headings{\let\@mkboth\markboth
  \def\@oddfoot{}
  \def\@evenfoot{}
  \def\@evenhead{\protect\underline{
    \hbox to\hsize{\bf \thepage\hfil \leftmark}}}
  \def\@oddhead{\protect\underline{
    \hbox to\hsize{\bf \rightmark \hfil\thepage}}}
  \def\chaptermark##1{\markboth {\ifnum \c@secnumdepth>\m@ne
    \@chapapp\ \thechapter. \ \fi ##1}{}}
  \def\sectionmark##1{\markright{\ifnum \c@secnumdepth >\z@
    \thesection. \ \fi ##1}}}
\makeatother
```

Listing 2

2	**Kapitel 1. Kapiteltitel**
1.1. Titel vom Abschnitt	**3**

Bild 2

Etwas Ausgefallenes gewünscht?

Die Definitionen der Makros, die in LaTeX benutzt werden können, sind über verschiedene Dateien verstreut. Ganz "unten" liegt der LaTeX-Kern, der vorcompiliert als Format in LPLAIN.FMT oder LPLAING.FMT eingelesen wird. Darüber stehen die verschiedenen Dokumentenstile wie *article* oder *book*. Für sie gibt es jeweils eine Stil-Datei als ASCII, also ARTICLE.STY oder BOOK.STY. Verwendet man Optionen, kommen noch weitere Dateien hinzu, beispielsweise TWOSIDE.STY. Bei den Dokumentenstilen gibt es noch weitere zusätzliche Optionsdateien, so zu BOOK.STY die Datei BK11.STY, die LaTeX dann liest, wenn man die Option 11pt angegeben hat. Standardmäßig wird übrigens immer zusätzlich BK10.STY geladen, was für die anderen Dokumentenstile analog gilt.

Die entscheidende Definition des Seitenstils *headings* ist in BOOK.STY zu finden (oder ARTICLE.STY und REPORT.STY und was es inzwischen noch geben mag). Es handelt sich um das Makro *\ps@headings*, das wie in Listing 1 aussieht. LaTeX übersetzt ein Kommando *\pagestyle{headings}* in einen Aufruf *\ps@headings*, bei *\pagestyle{empty}* würde *\ps@empty* aufgerufen.

In diesen *\ps@...*-Definitionen werden nun weitere Makros definiert; davon sollen uns zunächst nur *\@oddfoot*, *\@evenfoot*, *\@oddhead* und *\@evenhead* interessieren. Wenn LaTeX eine Seite ausgeben will, muß es noch die Kopf- und Fußzeilen dem eigentlichen Text hinzufügen. Dazu ruft es auf geraden Seiten die Makros *\@evenhead* und *\@evenfoot*, auf ungeraden *\@oddfoot* und *\@oddhead* auf. Aus der Definition dieser Makros ergibt sich also direkt der Inhalt der Kopf- und Fußzeilen. Ein Aufruf *\ps@...* ändert daher lediglich deren Definitionen auf den neuen Seitenstil.

Wenn wir nun diese Definitionen wiederum ändern, schaltet *\ps@headings* auf ein anderes Aussehen der Kopf- und Fuß-

```
\makeatletter
\def\ps@headings{\let\@mkboth\markboth
  \def\@oddfoot{}
  \def\@evenfoot{}
  \def\@evenhead{\protect\underline{
    \hbox to\hsize{\bf \thepage\hfil \leftmark}}}
  \def\@oddhead{\protect\underline{
    \hbox to\hsize{\bf \rightmark \hfil\thepage}}}
  \def\chaptermark##1{\markboth {##1}{}}
  \def\sectionmark##1{\markright{\ifnum \c@secnumdepth >\z@
    \thesection \ \fi ##1}}}
\makeatother
```

Listing 3

2 **Kapiteltitel**

1.1 Titel vom Abschnitt **3**

Bild 3

zeilen um - es sind also die vier genannten Makrodefinitionen innerhalb von *\ps@headings*, in die wir eingreifen müssen.

Wir fangen mit den geraden Seiten an und ändern die Kopfzeilen so, wie sie im LaTeX-Buch erscheinen. Dort steht ganz links die Seitennummer und rechts der laufende Titel, der sich hier aus der Kapitelüberschrift ergibt. Die Kommandos, die bei einem neuen Kapitel ausgeführt werden, legen diese Kapitelüberschrift in dem Makro *\leftmark* ab. Also könnte die Makrofolge

```
\thepage\hfil\leftmark
```

das Gewünschte erreichen. *\thepage* liefert die Seitennummer, *\hfil* ist ein wachsender Leerraum, der die beiden Texte nach außen "schiebt", und *\leftmark* ist, wie beschrieben, der Text in Abhängigkeit von der Kapitelüberschrift. Im LaTeX-Buch sind die Kopfzeilen in Fettschrift, also sollte noch ein *\bf* davorgesetzt werden.

Doch diese Definition funktioniert nicht, denn \hfil schiebt alles an den Rand der umgebenden Textbox. Eine solche gibt es hier aber nicht, man könnte genausogut *\thepage \leftmark* schreiben. Um die Definition herum muß eine \hbox mit einer bestimmten Breite, genauer: mit Seitenbreite, stehen. Die Kommandos

```
\hbox to\hsize{\bf\thepage\hfil\leftmark}
```

erzeugen jetzt einen Textkasten in Seitenbreite, in dem ganz links die Seitennummer steht und der Text nach rechts geschoben wird. Das entspricht aber noch nicht dem Gewünschten, denn das ganze soll ja noch unterstrichen sein. Unterstreichen kann man mit *\underline{<zu unterstreichender Text>}*. Der zu unterstreichende Text wäre die vorher beschriebene Textbox.

\underline ist allerdings im LaTeX-Buch als eines der "fragilen" Kommandos genannt, und in solchen Fällen sollte man besser ein *\protect* davorstellen.

Für ungerade Seiten sieht die Definition identisch aus, nur daß Seitennummer und Text vertauscht sind und *\rightmark* von anderen Kommandos in Abhängigkeit mit der Abschnitt-Überschrift (von *\section*) gesetzt wird. Listing 2 zeigt die geänderten Makros, die sich wie in Bild 2 auswirken.

Wohin aber mit den Makros? Sie sollten nicht die auf Ihrer Installation vorhandenen BOOK.STY, ARTICLE.STY oder REPORT.STY ändern. Schreiben Sie das Listing nach dem *\documentstyle*...-Kommando in Ihr LaTeX-Dokument. Daß das "at-sign" (@) in den Makronamen vorkommt, darf Sie nicht stören - zwischen *\makeatletter* und *\makeatother* ist dies erlaubt.

Etwas genauer

Genaugenommen haben wir aber noch nicht exakt das Aussehen des LaTeX-Buches erreicht. Dort steht auf geraden Seiten nur der Kapiteltitel, und zwar ohne "Kapitel x." oder "Chapter x" (ersteres, wenn Sie die *german*-Option verwenden). Offenbar kommt diese Markierung von *\leftmark*, denn in unserer Definition ist davon nicht die Rede. Auf ungeraden Seiten steht nach der Abschnittnummer ein Punkt. Das ist nach deutschsprachigen Regeln durchaus korrekt, wir wollen aber das Aussehen des englischsprachigen LaTeX-Buches exakt nachbilden. Dieser Punkt kommt von *\rightmark*.

Wie oben schon angemerkt, setzen die Kommandos bei Beginn eines Kapitels und von Abschnitten *\leftmark* und *\rightmark*. Wir haben sogar schon vor uns, wie sie das tun: jeweils durch einen Aufruf von *\chaptermark{<Kapitelüberschrift>}* beim Kapitelanfang und von *\sectionmark{<Abschnittüberschrift>}* beim Beginn eines Abschnitts. Und *\ps@headings* enthält genau diese beiden Definitionen. Wenn wir diese ändern, ändern wir das Aussehen von *\leftmark* und *\rightmark*.

Im Listing finden Sie übrigens *##1*. *#1* bezeichnet bei Makros ja immer den ersten Parameter, der hier in geschweiften Klammern übergeben wird. Wenn nun innerhalb eines Makros ein anderes mit Parametern definiert werden soll, müssen dessen Parameter anders bezeichnet werden. Man hängt ein weiteres # davor, und TeX weiß wieder exakt, was gemeint ist.

In *\chaptermark* steht ein länglicher Ausdruck*\ifnum \c@secnumdepth> \m@ne \@chapapp\ \thechapter. \ \fi*.

Dies sieht kompliziert aus und heißt natürlichsprachlich übersetzt: "Wenn Kapitelnummern erwünscht sind, füge *\@chapapp\ \thechapter. * ein". *\@chapapp* ist nur bei einer german-Option vorhanden; ansonsten steht dort "Chapter". *\@chapapp* ergibt "Kapitel", und *\thechapter* ist die Kapitelnummer. Und das sind genau die zwei Dinge, die noch stören, also werden sie einfach gelöscht, und die Definition von *\ps@headings* sieht wie in Listing 3 aus. Bei *\sectionmark* wird nur der überschüssige Punkt gelöscht.

Damit erhalten wir Kopfzeilen wie in Bild 3 und haben damit exakt dasselbe Aussehen wie im LaTeX-Buch.

Now for something different

Nun eine kleine Variation dazu: Die Kopfzeilen sollen nicht mehr wie im Buch von Leslie Lamport, sondern wie in der - übrigens rundum empfehlenswerten - "Einführung in TeX" von Norbert Schwarz [3] aussehen.

Dort sind die Kopfzeilen kursiv gesetzt, die Seitennummer steht jeweils außen, auf

linken Seiten steht die Kapitelüberschrift, rechts die Überschrift des Abschnitts. Diese beiden sind jeweils zentriert.

In den Definitionen muß natürlich in *\@evenhead* und *\@oddhead* eingegriffen werden. Es ändert sich die Darstellung innerhalb der *\hbox* (Listing 4). Mit \it wird jeweils die Kursivschrift angewählt, und durch *\hfil's* zentriert man die Texte in der Kopfzeile. Die Seitennummern stehen am selben Ort, nämlich auf der linken Seiten ganz links, auf der rechten Seite rechts außen. Im Gegensatz zum Stil des LaTeX-Buches ist die Kapitelnumerierung wieder vorhanden, dementsprechend fügt \thechapter. diese Zahl samt einem Punkt ein. In Bild 4 sind die Kopfzeilen abgedruckt - wenn Sie sie mit dem Buch vergleichen, sehen Sie die Übereinstimmung sofort.

```
\makeatletter
\def\ps@headings{\let\@mkboth\markboth
  \def\@oddfoot{}
  \def\@evenfoot{}
  \def\@evenhead{\protect\underline{
    \hbox to\hsize{\it \thepage\hfil \leftmark\hfil}}}
  \def\@oddhead{\protect\underline{
    \hbox to\hsize{\it\hfil \rightmark \hfil\thepage}}}
  \def\chaptermark##1{\markboth {\ifnum \c@secnumdepth>\m@ne
    \thechapter. \ \fi ##1}{}}
  \def\sectionmark##1{\markright{\ifnum \c@secnumdepth >\z@
    \thesection \ \fi ##1}}}
\makeatother
```

Listing 4

2 *1. Kapiteltitel*

1.1 Titel vom Abschnitt *3*

Bild 4

Und nun auch die Fußzeilen

Schließlich geht es nun noch um die Fußzeilen. Die Makros *\@oddfoot* und *\@evenfoot* zeigen dem anglophilen Leser schon an, daß es Pendants zu dem Definitionen der Kopfzeilen gibt. Und genau: In diesen beiden Makros werden die Fußzeilen festgelegt. Standardmäßig sind sie als leer definiert; man kann aber durchaus beliebige Definitionen in ihnen unterbringen. Die Positionierung der Fußzeilen läßt sich durch die Dimensionen *\footheight* (vertikaler Platzbedarf der Fußzeile) und *\footskip* (vertikaler Abstand zwischen der letzten Textzeile und dem unteren Rand der Fußzeile) festlegen.

In Listing 5 sind diese Definitionen nicht mehr leer, sondern mit Angaben über das Ausdruckdatum gefüllt. In einem Probeausdruck sollen so das Datum und die Zeit der Erstellung auftauchen. Auf ungeraden Seiten - also rechts - steht zentriert das Datum, das *\today* erzeugt. Links soll die Ausdruckzeit (genauer: die Uhrzeit des TeX-Laufs) stehen. Dazu verwendet die Definition im Listing eine Anwendung der Makros von Pierre A. MacKey, die in der Eingabe FONTBL.TEX auf Entwürfen von Roger L. Beeman basieren. *\currenttime* ergibt dabei die aktuelle Zeit im Format <Stunde>:<Minute>. Das Ergebnis sehen Sie in Bild 5, in dem wieder zwei Seiten zusammengefaßt sind.

```
\newcount\bighand\newcount\littlehand
\bighand=\time\divide\bighand by 60
\littlehand=\bighand\multiply\littlehand by -60
\advance\littlehand by\time
\def\currenttime{\the\bighand:\ifnum\littlehand<10{0}\fi\the\littlehand}
\makeatletter
\def\ps@headings{\let\@mkboth\markboth
  \def\@oddfoot{\hfil Ausdruck vom \today\hfil}
  \def\@evenfoot{\hfil\currenttime\hfil}
  \def\@evenhead{\protect\underline{
    \hbox to\hsize{\it \thepage\hfil \leftmark\hfil}}}
  \def\@oddhead{\protect\underline{
    \hbox to\hsize{\it\hfil \rightmark \hfil\thepage}}}
  \def\chaptermark##1{\markboth {\ifnum \c@secnumdepth>\m@ne
    \thechapter. \ \fi ##1}{}}
  \def\sectionmark##1{\markright{\ifnum \c@secnumdepth >\z@
    \thesection \ \fi ##1}}}
\makeatother
```

Listing 5

4 *1. Kapiteltitel*

Text...

0:50

1.1 Titel vom Abschnitt *5*

Text...

Bild 5

Wie benutzen?

Die bisher vorgestellten Makros sind bei der Verarbeitung eines LaTeX-Dokuments noch eher schlecht eingebunden, weil sie in jedem Dokument Zeile für Zeile auftauchen müssen. Damit man sie nicht jedesmal neu eintippen muß, könnte man sie in einer Datei HEADFOOT.TEX in einem TeX-Eingabeverzeichnis ablegen und jeweils durch *\input headfoot* einbinden. Das würde aber nicht recht in die LaTeX-Konzeption passen. Dort werden alle Dateien, die Makros verändern, schon in der ersten Zeile des Eingabetextes festgelegt, und zwar mit *\documentstyle [<Optionen>]{<Stil>}*. *<Stil>* ist dabei eine grundlegende Klassifizierung eines

Schriftstückes, beispielsweise eines Artikels (als *article*) oder eines Buchs (als *book*). Die <*Optionen*> legen verschiedene Veränderungen, Erweiterungen oder Abwandlungen des Dokumenten-Stils fest.

Bei der Verarbeitung lädt LaTeX zunächst eine Stil-Datei, die der Definition von <*Stil*> entspricht (also bei *\documentstyle{article}* die Datei ARTICLE.STY) und danach weitere Dateien entsprechend der Optionen in deren Reihenfolge (also bei *\documentstyle [dina4,german] {article}* zusätzlich die Dateien DINA4.STY und GERMAN. STY).

Änderungen und Erweiterungen plaziert man unter LaTeX also in einer Datei mit der Endung .STY, die auf Wunsch geladen wird. In unserem Fall kommt das abgedruckte Listing nach Abtippen also in eine Datei HEADFOOT.STY, die in dem Verzeichnis abgelegt wird, in dem Ihre TeX- bzw. LaTeX-Installation nach den Styles sucht. Damit sie bei der Verarbeitung des Dokuments auch eingelesen wird, muß sie als Option im *\documentstyle*-Befehl auftauchen, also beispielsweise

```
\documentstyle[dina4,german,headfoot]{article}
...
```

Es ist zu empfehlen, *headfoot* als letzte Option zu verwenden. Sollte danach eine Option aufgeführt werden, die dieselben Definitionen enthält, werden die aus *headfoot* im Verarbeitungsprozeß überschrieben.

Während der Verarbeitung der Options-Dateien ist das Zeichen "@" übrigens noch als Buchstabe in Kommandonamen erlaubt - die Zeilen mit *\makeatletter* und *\makeatother* können in HEADFOOT. STY also entfallen.

Finit

Für die Veränderungen an den Kopf- und Fußzeilen in LaTeX ließen sich noch weitergehende Makros schreiben. Man könnte auch neue Seitenstile definieren. Sollte ein Seitenstil kopffuß implementiert werden, muß ein entsprechendes Makro *\ps@kopffuß* vorhanden sein, das in analoger Weise dieselben Makros festlegt. Um eine Kompatibilität zu anderen LaTeX-Systemen zu erhalten, sollten Sie nicht unbedingt neue Seitenstile einführen (auch wenn Leslie Lamport das in dem kommentierten Quelltext von LaTeX anregt).

Solche Makros und ihre sichere Implementierung würden den Rahmen dieses Artikels sprengen, der sich an den "normalen" LaTeX-Benutzer wendet. Abschließend darf ich Sie noch auf den Kasten für TeX-Benutzervereinigungen in der BRD und weltweit verweisen - in deren Publikationen finden sich auch die Anregungen für TeXniker und TeXacker!

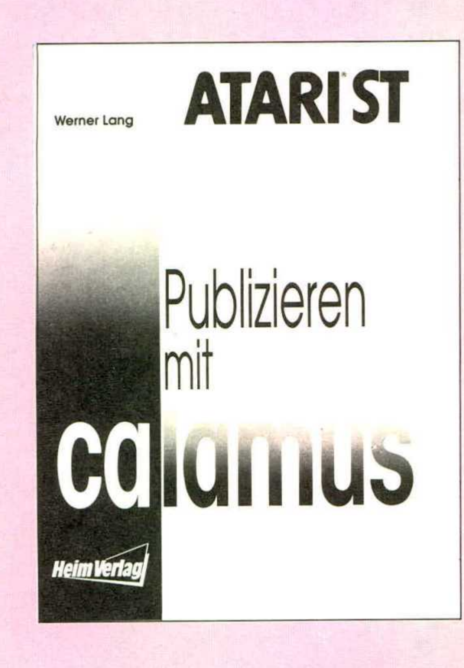
RT

[1] Leslie Lamport
LaTeX: A Document Preparation System
5. Auflage, Reading/Mass. 1986
242 Seiten
ca. DM 58,- ISBN 0-201-15790-X

[2] Donald E. Knuth
The TeXbook
11. Auflage, Reading/Mass. 1987
Addison-Wesley
483 Seiten
ca. DM 74,- ISBN 0-201-13448-9

[3] Norbert Schwarz
Einführung in TeX
2. Auflage, Bonn, 1988
Addison-Wesley-Verlag
272 Seiten
DM 68,- ISBN 3-925118-97-7

Neuheiten



Best. Nr. B-442
ISBN-Nr. 3-923250-00-00
Ca. 300 Seiten
Hardcover

49,- DM
unverb. empfohlener Verkaufspr.

Das Publizieren vom Schreibtisch aus stellt die konsequente Weiterentwicklung der Textverarbeitungsprogramme dar. Desktop-Publishing wurde zum Schlagwort am Softwaremarkt.

Das Softwarepaket „**CALAMUS**" bietet für den ATARI ST die Voraussetzung für den Einstieg in die professionelle Setztechnik. Wie kein anderes Programm verwirklicht es das

„**what you see is what you get**"-Prinzip.

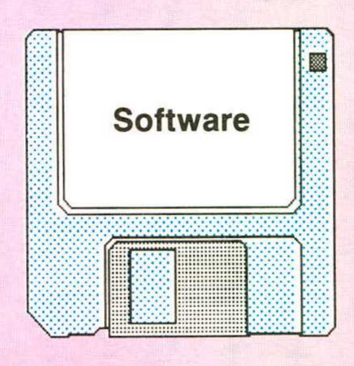
Die Druckqualität stellt eine Herausforderung an die Ausgabeeinheit dar.

Die kreative Arbeit wird nur zum Erfolg, wenn der Anwender sinnvolle Vorgaben in das Programm einbringt. Ein umfangreiches Vorwissen ist nötig, um mit dem Programm arbeiten zu können.

Dieses Buch ist ein unentbehrliches Hilfsmittel für alle EINSTEIGER, UMSTEIGER und AUFSTEIGER in das weite Feld des Desktop-Publishing.

In den einzelnen Kapiteln werden Sie mit den Möglichkeiten und den Problematiken der elektronischen Erstellung von Druckvorlagen vertraut gemacht. Viele praxiserprobte Tips erleichtern Ihnen den täglichen Umgang mit dem Softwarepaket „Calamus".

Wenn Sie bisher mit Calamus keine befriedigenden Ergebnisse erzielen konnten, sollten Sie nicht länger auf dieses Buch verzichten.



pro-CAM

Life-Simulation

189,-

Im Gegensatz zu herkömmlichen Simulationskonzepten gibt es eine neue Idee namens CAM. Cellular Automata Machines arbeiten auf rein graphischer Ebene. Sie sind heute so weit entwickelt, daß sie komplexe naturwissenschaftliche Vorgänge simulieren können.

Erstmalig ist nun ein professioneller Cellulärer Automat (CAM) für den ATARI erhältlich. Er bietet eine graphische Benutzerschnittstelle und ist daher gut geeignet, den Einstieg in diese neuartige Welt der Simulationssysteme zu vollziehen. Mit dem System können komplexe Probleme bearbeitet werden: Zu dem weitgefächerten Anwendungsspektrum von Cellulären Automaten gehört u.a. Simulationen von chemischen Reaktionen und physikalischen Prozessen. Ebenso können Fragestellungen der Ökonomie, der Soziologie und der Biologie behandelt werden.

Eigenschaften von pro-CAM:

- benutzerfreundliche, mausgesteuerte Arbeitsoberfläche
- hohe Geschwindigkeit (bis zu 1 Generation pro sek bei einer Auflösung von 256 x 160 Pixel), da vollständig in Assembler programmiert
- Filmoption erlaubt bis zu 8 Generationen pro sek
- Farb- und S/W-Versionen sind eigenständige Programme mit jeweiligen Stärken
- Lieferumfang: 1 Buch, 1 Diskette mit Programmen, 4 Disketten mit Anwendungen aus Mathematik, Physik, Chemie, Biologie und Geographie
- bereits bei mehreren europäischen Universitäten im Einsatz

unverbindlich empfohlene Verkaufspreise

BESTELL-COUPON an Heim-Verlag Heidelberger Landstraße 194 6100 Darmstadt-Eberstadt

Ich bestelle: ______ DTP mit CALAMUS à 49,— DM
______ pro-CAM, Life-Simulation à 189,— DM
zuzügl. Versandkosten DM 6,– (Ausland DM 10,–) unabhängig von der bestellten Stückzahl
[ ] per Nachnahme [ ] Verrechnungsscheck liegt bei

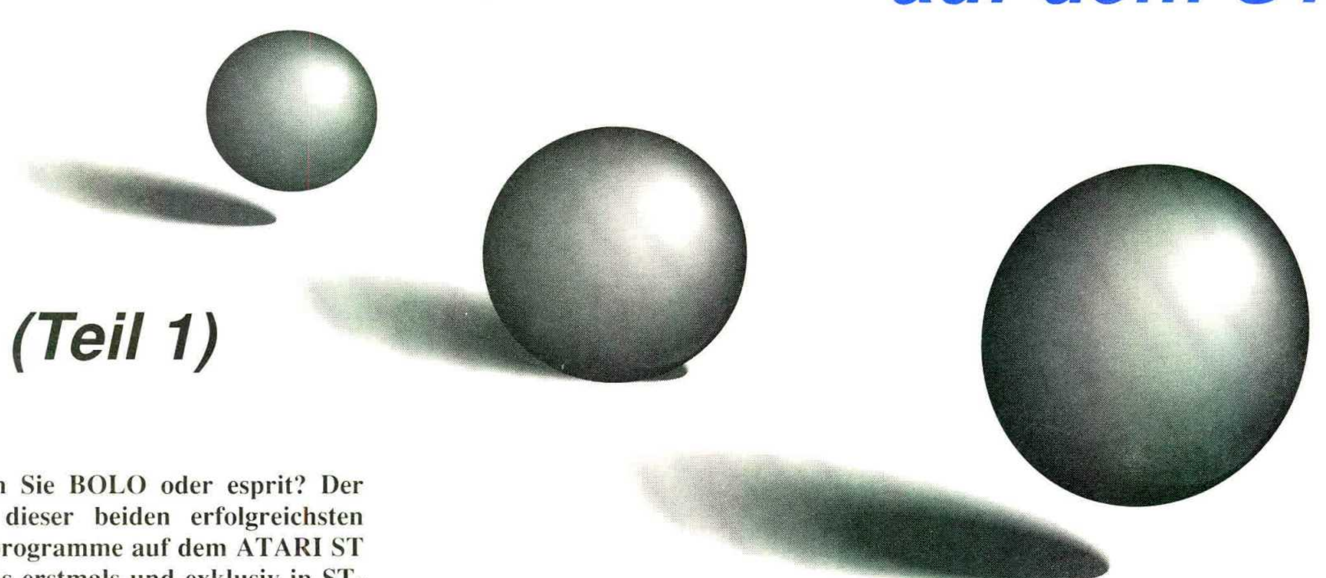
Name, Vorname ____________________
Straße, Hausnr. ____________________
PLZ, Ort ____________________
Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Schweiz
Data Trade AG
Landstr. 1
CH - 5415 Rieden - Baden

Österreich
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

Professionelles Spiele-Design auf dem ST



(Teil 1)

Kennen Sie BOLO oder esprit? Der Autor dieser beiden erfolgreichsten Spieleprogramme auf dem ATARI ST läßt uns erstmals und exklusiv in ST-Computer hinter die Kulissen einer professionellen Spielentwicklung schauen.

Spielen ist die schönste Nebensache der Welt. Und seitdem es Computer gibt, gibt es auch Computerspiele. Und immer mehr erwachsene Menschen entdecken das Spiel als eine angenehme Freizeitgestaltung. Dabei haben Computer neben den traditionellen Gesellschaftspielen (die mit den Würfeln und kleinen Holzfiguren) völlig neue Spielmöglichkeiten geschaffen. Eine wesentliche Neuerung ist die Möglichkeit des Spielers, den Spielverlauf in einer direkten Interaktion zu beeinflussen. Eine andere Neuerung ergibt sich bei den Spielregeln. Diese müssen einem Computerspiel ja mit einprogrammiert werden, wodurch der Computer dann auch als Schiedsrichter fungiert. Gute Computerspiele nutzen diesen Umstand auch dazu, dem Spieler während des Spielens die Spielregeln beizubringen. Darüber hinaus erhalten Computerspiele durch hübsche Animationen und realistische Geräusche ihren besonderen Reiz. Alles Möglichkeiten, die ein normales Brettspiel nicht bieten kann.

Aber wie entsteht ein solches Computerspiel? In diesem Artikel geht es darum, einmal aufzuzeigen, welche Kriterien die Qualität einer Spielidee bestimmen und welche Probleme bei der Konzeptionierung und Programmierung eines Spiels anfallen.

Aller Anfang ist schwer

Die Entwicklung eines Computerspiels ist eine langwierige Angelegenheit, die sich grob in fünf Entwicklungsstufen einteilen läßt:

1. Entwicklung der Spielidee
2. Erstellung eines Konzepts zur Realisierung der Spielidee
3. Kodierung des Konzepts
4. Erstellung der einzelnen Spielbilder
5. Austesten und Ausfeilen des fertigen Spielprogramms

Am Beginn eines Projektes steht die Spielidee. Eine Idee besteht aus der Vorstellung dessen, was später einmal auf dem Bildschrim und im Kopf des Spielers passieren soll. Davon begleitet sind eine Unmenge von Bildschirmskizzen, Tagträumereien und - nach Möglichkeit - massig viele Gespräche mit ähnlich verspielten Leuten. Die Entwicklung einer Idee kann dann schon - neben den anderen täglichen Arbeiten - einige Monate dauern. Immer wieder werden Ideen abgeändert oder gar verworfen, bis eine Spielidee vorliegt, die meine fünf Regeln des Spiele-Designs erfüllt:

1. Ein Spiel soll keine brutale Gewalt darstellen.
2. Der Computer sollte als "Spielwiese", nicht als Gegner dienen.
3. Die Steuerung muß einfach und präzise sein.
4. Die Spielregeln sollten intuitiv während des Spielens erlernt werden können.
5. Das Spiel muß auch ungeübten Spielern Spaß bringen.

Regel Nr.1 ist ein rein persönliches Anliegen. Zu Regel Nr.2 ist anzumerken, daß hierfür die Ideallösung das "Zweiund-mehr-Spieler"-Spiel darstellt. In der Praxis sind solche Spiele aber kaum zu vermarkten, weshalb oft das "Ein-Spieler"-Spiel bevorzugt wird. Wenn man also ein solches programmieren möchte, sollte man darauf achten, daß der Spieler gegen sich SELBST spielt. Dies ist eine wichtige psychologische Feststellung, die die Motivation eines Spiels stark bestimmt. Probieren Sie es doch selbst einmal mit BOLO aus! Haben Sie schon festgestellt, daß es immer SIE sind, die (der) schuldig im Sinne der Anklage ist, wenn der Ball die untere Grenze des Bildschirms überflogen hat? Es war nicht(!) der Computer Sie selbst waren es!

Bei BOLO ist das auch der Grund, warum die erzielten Punktzahlen eigentlich keine große Rolle spielen. Deshalb wurde der Punktfetischismus bei esprit auch gleich ganz weggelassen. Stattdes-

sen ist hier die Möglichkeit geschaffen worden, schon gespielte Bilder bei einem neuen Spielanfang zu überspringen.

Spiele, die man alleine oder zu mehreren spielen kann, gehören in den meisten Fällen zu der Kategorie "Zwei-Spieler-Spiel". Spielt man ein solches Spiel alleine, so wird der Part des zweiten Spielers vom Computer übernommen. Damit erfüllt dieser Spieltyp nicht die Regel Nr.2.

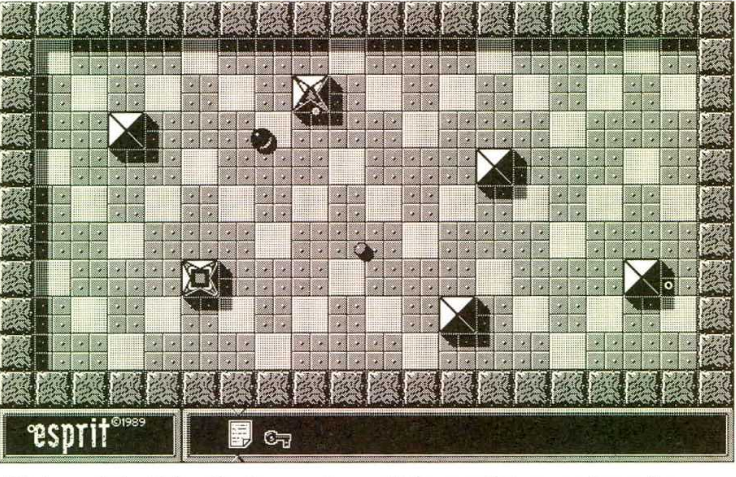


Abb.1: esprit in Aktion. Soeben wurden zwei Memory-Steine von der großen, schwarzen Spieler-Kugel berührt. Aber Vorsicht! In diesem Bild wird die kleine Killerkugel ständig in Richtung zur großen beschleunigt.

Regel Nr.3 sollte eigentlich jedem sofort einsichtig sein, trotzdem wird sie in der Praxis unverständlicher Weise oft nicht beachtet. Man kann sich die Realisierung der besten Spielidee sparen, wenn die Steuerung hinterher nicht richtig funktioniert. Ideal sind die Steuerungen, die den natürlichen Instinkt des Spielens unterstützen, wie z.B. das Dribbeln eines Balles in der realen Welt. Dieser Analogie zur natürlichen Welt kommt man übrigens mit einem analogen Eingabegerät, wie einem Paddle oder einer Maus, am besten nach. Digitale Joysticks schränken den Computerspieler in seinen Handlungsmöglichkeiten stark ein.

Die Regel Nr.4 hängt wesentlich von der Art des Spiels ab. Mir sind jedoch die Spiele am liebsten, bei denen ich keine hundertseitigen Handbücher durchwühlen muß.

Regel Nr.5 ist eigentlich eine Frage des späteren Bilder-Designs, muß aber schon in der Spielidee berücksichtigt werden. Diese Regel wird ebenfalls bei vielen Spielen nicht beherzigt. Nur wahre Joystick-Artisten haben bei den meisten Spielen die Chance, das erste Bild zu überstehen. Man sollte dies aber auch dem ungeübten Spieler ermöglichen. Wichtig ist hier der richtige Verlauf der Schwierigkeitskurve vom ersten zum letzten Bild. Klar, daß auch Joystick-Artisten auf ihre Kosten kommen sollen. Wenn jedoch jemand, der nicht so geübt ist, hundert Mark für ein Spiel ausgibt, sollte er doch erwarten können, daß er den größtmöglichen Spielspaß eines Spiels für sein Geld erhält und nicht nach den ersten Bildern frustriert das Handtuch werfen muß.

Hat man nun eine schöne Idee vor dem geistigen Auge, geht es an die Konzeptionierung dieser Idee, bei der die technischen und grafischen Realisierungsmöglichkeiten abgetestet werden.

Die Konzeptionierung eines Spieles unterscheidet sich dabei grundsätzlich von allen anderen Projekten: Normalerweise ist immer ein Problem vorgegeben, das es zu lösen gilt. Bei Spielen schafft man sich jedoch die Probleme selbst! Das ist einerseits ein Vorteil, andererseits ein Nachteil. Der Vorteil ist, daß man ggf. ein Problem entschärfen kann, um es zu lösen. Der Nachteil ist, daß man sich die Meßlatte doch möglichst hoch legen will und die Probleme sich meist von der Grundidee ableiten, also oft gar nicht richtig zu entschärfen sind. Gute Spieleprogrammierer zeichnen sich demnach durch extremen Masochismus aus. Dies ist aber auch notwendig, um die durch die eigene Idee vorgegebene technische Hürde nehmen zu können.

```
...
StopLevel := FALSE;
InitLevel;
REPEAT
  REPEAT UNTIL SwitchOk;
  (* SwitchOk wird vom VBL-
   * Interrupt auf TRUE gesetzt *)
  RedrawBackground;
  DrawFigures;
  RedrawBackgroundStones;
  SwitchSides;
  (* Hiermit wird auch SwitchOk
   * auf FALSE gesetzt *)
  MoveFigures;
  AnimateBackground;
UNTIL StopLevel;
...
```

Abb.2: Hauptschleife von esprit zur Steuerung der Animationen und des Bildaufbaus.

Die Spielidee von esprit könnte folgendermaßen formuliert werden:

"esprit (Abb.1) liegt die Idee zugrunde, eine komplette Computerwelt mit physikalischen Effekten zu simulieren, in der der Spieler möglichst frei durch eine mausgesteuerte Kugel agieren kann. Als Basis muß das gute alte Memory-Spiel herhalten, allerdings um eine entscheidende Komponente erweitert: Um an einzelne Memory-Steine heranzukommen, muß der Spieler die Regeln und Tücken der simulierten Welt herausfinden und anwenden. Die eigentliche Welt unterteilt sich in drei Ebenen. Jede Ebene ist in eine Matrix kleinerer Felder unterteilt. In der untersten Ebene sind die sog. "Bodenplatten" angeordnet. Der jeweils in einem Feld gesetzte Bodenplattentyp definiert die Reibung und die Beschleunigung der darauf befindlichen Figuren. Die darüber liegende Ebene enthält Objekte, die die Figuren aufnehmen und an anderer Stelle wieder ablegen können. Die oberste Ebene ist die Steinebene, die die Bewegungsfreiheit der Figuren bestimmt. Was das Spiel hinterher interessant machen soll, ist die gegenseitige Beeinflußung der verschiedenen Steine, Objekte und Figuren."

Man muß sich vor dem Programmieren der ersten Zeile genau über die technischen und konzeptionellen Probleme im klaren sein. Im folgenden möchte ich Ihnen ein paar ausgewählte Probleme aufzeigen. Es handelt sich zum Teil um Probleme, die bei allen Spielentwicklungen auftreten, teils um spezielle Probleme, die bei der Entwicklung der Spiele esprit und BOLO autraten.

Zwei-Seiten-Grafik

Heutzutage verwenden alle Action-Spiele eine Zwei-Seiten-Grafik. Man benötigt zwei Seiten, damit der Bildaufbau unsichtbar bleibt und ein Flackern des Bildes vermieden wird. Das Pinzip ist einfach: Die eine Grafikseite wird angezeigt, während auf der anderen das neue Bild gezeichnet wird. Anschließend wird das neu aufgebaute Bild dargestellt und das nächste Bild auf dem zuvor dargestellten neu gezeichnet (Abb.2). Bei esprit geschieht dieser Vorgang 72mal in der Sekunde. Beim Monochrommonitor des ATARI ist dies die vorgegebene Bildwechselfrequenz. Nur wenn man es schafft, das Bild innerhalb einer 72tel Sekunde aufzubauen, erhält man den besten Animationseindruck. Ist man während des Bildaufbaus auch nur einen Zyklus zu spät, sinkt die softwaremäßige Bildfrequenz auf die Hälfte, also 36 Bilder pro Sekunde, ab. In diesem Fall würde auf dem Monitor jeweils zweimal nacheinander ein und dasselbe Bild erscheinen. Man möchte es

nicht glauben, aber man muß diesen Unterschied einmal gesehen haben: Der Eindruck einer echten 72Hz-Animation ist um ein vielfaches realistischer und angenehmer. (Probieren Sie es doch selbst einmal z.B. mit dem Creator und seinen Titeleffekten aus!)

72 Bilder pro Sekunde zu generieren, ist verdammt schwierig. Farbspiele haben es da wesentlich einfacher. Dort braucht man "nur" 50 Bilder pro Sekunde zu erzeugen, um eine optimale Animation zu erhalten. Nun ist klar, daß es unmöglich ist, jedesmal innerhalb einer 72tel Sekunde ein Bild komplett neu zu zeichnen. Was man ergo benötigt, ist eine Verwaltung, die in dem nicht dargestelltem Bild nur die Bereiche

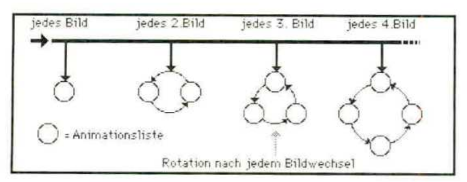


Abb.3: Verkettung der Animationslisten zur Animationsoptimierung.

ändert, die sich zwischen den beiden(!) letzten Bildwechseln tatsächlich verändert haben. Zu diesem Zweck sind die beiden Bildschirmseiten in 32 mal 32 Pixel große Felder (Longword-Zugriff!) unterteilt. Immer, wenn eine Routine im System etwas auf den Bildschirm zeichnet, wird das betreffende Feld markiert. Nachdem ein Bild dargestellt worden ist, werden die so gekennzeichneten Bereiche auf dem jeweiligen Bildschirm restauriert.

Animationsoptimierung

Ein zeitkritisches Problem besteht darin, daß für viele Objekte und Steine im Hintergrundbild in regelmäßigen Abständen eine Prozedur aufgerufen werden muß, die die Animation und andere regelmäßige Tätigkeiten des betreffenden Objekts oder Steins ausführt. Würde man bei jedem Bildaufbau für jedes Feld (maximal 2000 Felder!) eine Animationsprozedur aufrufen, so würde die Zeit nicht ausreichen. Die Alternative ist die, die Animationsprozedur für jedes Feld nur dann aufzurufen, wenn wirklich etwas passieren soll.

Was man also benötigt, ist eine optimierende Animationsverwaltung. Jedes zu animierende Feld (Stein oder Objekt) muß sich folglich - unter Angabe seiner Animationsprozedur und der Aufrufgeschwindigkeit (z.B. bei jedem dritten Bildwechsel) - bei dem Animationsmanager anmelden. Der Animationsmanager hat nun die Aufgabe, schnell und optimal bei jedem Bildwechsel alle so angemeldeten Felder durch Aufruf seiner Animationsprozedur zu versorgen. "Optimal" heißt hier, daß,

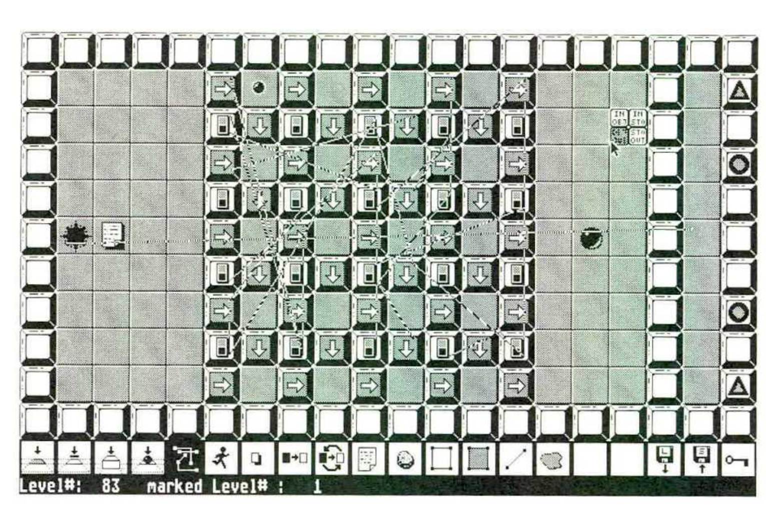


Abb.4: Mit dem esprit-Editor (nicht käuflich, Anfragen zwecklos!) wurden die einzelnen Bilder des Spiels kreiert. Die Linien auf dem Spielfeld repäsentieren die gelegten logischen Signalleitungen.

wenn sich z.B. drei Felder angemeldet haben, die jeweils nur nach drei Bildwechseln versorgt werden müssen, nach jedem Bildwechsel eines der Felder aufgerufen wird. Die notwendige Rechenzeit zur Animation der einzelnen Felder soll sich möglichst gut auf alle Bildwechsel verteilen. Aus diesem Grund existieren für jede Animationsfrequenz Animationslisten, in denen die zu animierenden Felder vermerkt sind. Für die Animationsfrequenz von 3 gibt es drei Listen, für die Animationsfrequenz von 10, die minimalste Frequenz bei esprit, gibt es 10 verschiedene Listen. Meldet sich nun ein Feld an, wird unter den Listen der jeweiligen Animationsfrequenz die Liste herausgesucht, die bisher die geringste Anzahl an Einträgen besitzt. In diese wird das neu angemeldete Feld angehängt.

Um die so aufgebauten Listen abzuarbeiten, sind alle Animationslisten einer Animationsfrequenz in jeweils einem Ring zusammengefaßt. Auf jede Ringliste zeigt ein globaler Zeiger. Nach jedem Bildwechsel wechseln die globalen Zeiger zur nächsten Animationsliste in der Ringliste, welche dann durch Aufruf aller in der Animationsliste befindlichen Animationsprozeduren abgearbeitet wird. Abb. 3 zeigt den Aufbau dieser Datenstruktur.

Auf den ersten Blick scheint diese Lösung kompliziert. Doch löst diese Listenstruktur das Problem der optimalen Animation, zumal sämtliche Änderungen, wie z.B. das An- und Abmelden von Feldern, durch einfaches Umlegen von Zeigern zu realisieren ist.

Wer vorher rechnet, ist hinterher schneller

Die grafische Darstellung ist eines der zeitkritischsten Probleme. Ein wesentlicher Trick, um eine möglichst hohe Geschwindigkeit zu erzielen, besteht darin, so viel wie nur möglich vorher zu berechnen, so daß man im zeitkritischen Moment nur noch auf die berechneten Daten zugreifen braucht. Dabei geht man allerdings einen Tausch zwischen Speicherbedarf und Geschwindigkeit ein. Je mehr Speicher zur Verfügung steht, desto mehr kann an zusätzlicher Geschwindigkeit herausgeholt werden. Ein klassisches Beispiel ist die Sprite-Darstellung:

Um eine Figur auf dem Bildschirm darzustellen, muß man diese zuvor auf einem anderen Bildschirm gezeichnet haben und von dort auf den sichtbaren Bildschirm kopieren. Dabei ergibt sich ein Problem, das mit dem Aufbau des Bildschirms zusammenhängt. Es reicht nicht aus, die einzelnen Bytes des einen Bildschirms in den anderen zu kopieren. Will man die Figur auf eine beliebige horizontale Position darstellen, müssen die einzelnen Bits, die ja die Bildpunkte repräsentieren, innerhalb eines Bytes verschoben werden. Und dies kann bei vielen zu zeichnenden Figuren für eine 72Hz-Animation zu langwierig sein. Die Lösung des Problems: Man berechnet vorher alle acht Möglichkeiten zur horizontalen Positionierung eines Zeichens. Das hört sich leicht an, erfordert jedoch einigen Aufwand, zumal man sich ein Programm schreiben muß, das hier entsprechende Listen mit "vorgeshifteten" Bildern erzeugt und verwaltet. (Auf den Spieledisketten sind dies bei mir die Dateien mit der Endung .SHL = Shapelist)

Ein Tausch von Speicher gegen Geschwindigkeit anderer Art entsteht bei der Benutzung von sog. Linearroutinen. Damit sind Routinen gemeint, die normalerweise aus einer Schleife bestehen, in der für jeden Schleifendurchlauf eine Berechnung aufgrund des Schleifen-Indizes durchgeführt wird. Eine solche Routine läßt sich linearisieren, indem man alle Schleifenduchläufe zuvor berechnet und die daraus resultierenden Befehle nacheinander im Speicher ablegt.

VORTEX TECHNOLOGIE UND ZUKUNFT

vortex ATonce - Der kleine Unterschied !
Und aus Ihrem ST oder Mega ST wird ein AT-kompatibler Computer.

Machen Sie aus Ihrem Atari ST oder Mega ST einen AT-kompatiblen Computer. Soweit die Atari ST Computer dies erlauben. vortex ATonce ist der AT-Emulator mit 80286 (16 Bit) Prozessor und 8MHz Taktfrequenz. Er bringt Norton Faktor 6.5. Das sind ca. 50 Prozent mehr als ein vergleichbares System mit V30-8MHz-Prozessor. Bei ST´s mit mind. 1 MB RAM stehen 704 KB DOS-Speicher zur Verfügung. DOS kann direkt von einer Partition geladen werden.

ATonce unterstützt die vortex Fest- und Wechselplatten-Systeme HDplus und DataJet sowie andere voll kompatible Fest- und Wechselplatten. ATonce emuliert diverse Grafik-Karten und Schnittstellen. Maus- und Sound-Unterstützung ist keine Frage. ATonce wird in den ST Computer eingebaut; in STE und Mega ST über Steckadapter. Damit Sie den kleinen Unterschied selbst feststellen können, haben wir ATonce hier in Originalgröße abgebildet.

Wollen Sie mehr über vortex ATonce wissen? Wir senden Ihnen gerne weiteres Informationsmaterial zu.





VORTEX COMPUTERSYSTEME GMBH . FALTERSTRASSE 51-53 . D-7101 FLEIN . TELEFON 07131 / 50 88-0
COMPUTERSYSTEME VORTEX AG . BUNDESPLATZ 3 . CH-6300 ZUG . TELEFON 042 / 21 84 42

Der esprit-Editor

Ein Spiel besteht in der Regel nicht nur aus dem später ausgelieferten Spielprogramm. Man benötigt eigentlich immer noch ein zweites, spezielles Programm, um die einzelnen Bilder eines Spiels zu edieren. Bei solchen Spieleditoren versucht man den Aufwand so gering wie möglich zu halten, was dann allerdings dazu führt, daß diese Programme nur vom Autor selbst zu bedienen sind. So auch bei esprit (Abb.4). Hier ist der Editor aufgrund der vielen Möglichkeiten recht komplex geworden. Hauptaufgabe des esprit-Editors ist das Setzen der Bodenplatten, der Objekte und der Steine. Man wählt z.B. seinen Steintyp und legt ihn durch einen Mausklick auf dem Spielfeld ab.

Eine andere wichtige Funktion des esprit-Editors ist das Legen von logischen Verbindungsleitungen zwischen Steinen und Objekten. Mit diesen Leitungen wird z.B. festgelegt, wo ein Transporterobjekt die Kugel hinteleportieren soll oder bei welchem Objekt oder Stein ein Schalterstein eine Reaktion auslösen soll. Es handelt sich hier quasi um binäre Leitungen, auf die die verschiedenen Objekte und Steine abhängig von ihrem Grundtyp reagieren. So reagiert ein normaler Stein auf das Signal "1" damit, daß er sich in einen Schweberstein verwandelt, unter dem die Kugel durchrollen kann. Ein Schweberstein wiederum reagiert auf das Signal "0" damit, daß er sich in einen normalen Stein verwandelt. Ein Schalterstein wiederum sendet bei Berührung der Kugel auf allen angeschlossenen Ausgabeleitungen entweder eine "1" oder eine "0", je nach Schalterstellung. Ein Schalterstein schaltet aber nicht nur bei Berührung mit der Kugel, sondern kann auch beim Empfang auf seiner Empfängerleiterung seine Schalterposition ändern. Damit ist es bei massiver Verdrahtung später im Spiel möglich, eine äußerst komplizierte Signalbewegung auf dem Spielfeld in Gang zu setzen.

Daneben gibt es noch eine Reihe weiterer Funktionen im esprit-Editor: das Testen von Bildern, das Laden und Speichern von Bildern und die Definition von allgemeinen Parametern eines Bildes, wie z.B. die Startpositionen der Figuren oder die Texte für die Notizzettel. Eine der wichtigsten Aufgaben aber ist die Programmierung der verschiedenen Objekt-, Stein- und Figurentypen:

Der MEGA-GHOST-Trick

Bei der Konzeptionierung von esprit war sofort klar, daß hier viele Spezialeffekte in Form des Verhaltens einzelner Stein- und Objekttypen zu programmieren waren. So gibt es z.B. einen Stein, der bei Berührung zehnmal ein Form-Feed und die Zeile "KÖLLE ALAAF!!" auf den Drucker ausgibt. Da es eine ganze Reihe von solchen Effekten und logischen Zusammenhängen geben sollte, kam eine "normale" Programmierung von Edieren, Compilieren, Linken und Probieren erst gar nicht in Betracht. Glücklicherweise beherrscht das Megamax-Modula-2-System das sogenannte "Loadtime-Linking". Megamax-Modula-2 verfügt über ein Modul, den "Loader", der, während man sich im Modula-System befindet, beliebige Moduln laden und mit den schon im Speicher befindlichen Moduln verbinden kann. Es weiß folglich immer, welche Programmteile gerade im Speicher sind. Mehr noch: Soll ein neues Modul hinzugelinkt werden, stellt es fest, welche weiteren Moduln für den Start des eigentlich zu linkenden Moduls benötigt werden und startet diese ebenfalls. Genauso wie man Moduln hinzulinken kann, können sie auch wieder entfernt werden.

Abb.6: Funktionsbild zum Mega-Ghost-Trick.

Und nun der Zaubertrick des Mega-Ghost: Alle allgemeinen Moduln, wie der Bildschirm- und der Sound-Treiber oder die Animationsverwaltung werden zu einem Programm zusammenfaßt, dem esprit-Editor. Die Spielsteuerung, die Spezialeffekte, die Prozeduren für die Steuerung der verschiedenen Stein-, Objekt- und Figurentypen werden in einzelnen Treibermoduln zusammengefaßt. Diese Treibermoduln sind normale Programmoduln und importieren zwar aus den übrigen Moduln des Systems, selbst exportieren sich diese nur dadurch, daß sie sich mit Prozeduren bei den anderen Moduln anmelden. Wird nun der esprit-Editor gestartet, werden die Treibermoduln nachgeladen, also an das System "angeklinkt". Der esprit-Editor besitzt nun selbst Funktionen, mit dem er den Editor des Megamax-Systems mit der automatischen Angabe des betreffenden Treibermodultextes aufrufen kann. Hier kann nun die gewünschte Änderung programmiert werden. Wird der Editor mit der Antwort "EXECUTE" verlassen, ruft der esprit-Editor den Megamax-Compiler auf und sorgt anschließend dafür, daß das betreffende alte Treibermodul entladen und das soeben neu übersetzte Modul wieder angelinkt wird. Da mit dieser Technik nur noch das Modul mit der notwendigen Änderung compiliert und gelinkt zu werden braucht, erzielt man einen enormen Zeitvorteil. Würde man immer wieder das gesamte Programm linken und starten müssen, weil man irgendwo eine kleine Änderung machen mußte, würde die Turn-Around-Zeit um ein vielfaches höher liegen, da dann für den esprit-Editor an die 20 Moduln geladen werden. Je nach Länge des zu bearbeitenden Treibermoduls verringert sich die Turn-Around-Zeit auf bis zu ca. 5 Sekunden. Außerdem entfällt bei dieser Methode das Laden vieler Dateien bei der notwendigen Initialisierung der Moduln. Dieser interessante Trick ist genauso gut für andere Projekte verwendbar. Abb.5 und Abb.6 skizzieren die allgemeine Anwendung dieses Tricks. Es gibt auch noch ein anderes Modula-2 für den ST (SPC-Modula), das einen Loadtime-Linker besitzt. Aber einzig und allein das Megamax-Modula-System erlaubt die Möglichkeit der Modulanbindung.

Soviel für heute. Nächsten Monat geht es im zweiten und letzten Teil u.a. um physikalische Effekte, Geräusche und Schattenspiele.

Meinolf Schneider

```
MODULE LinkTrick;

(* Erstellt mit Megamax-Modula-2.
 * (Vertrieb: Application Systems Heidelberg Software GmbH) *)

FROM    System          IMPORT  ADDRESS, ADR, BYTE;
FROM    Terminal        IMPORT  WriteString, Write, WritePg, Read, Writeln;
FROM    Strings         IMPORT  Append;
FROM    StrConv         IMPORT  CardToStr;
FROM    Paths           IMPORT  StdPaths, PathList;
FROM    Loader          IMPORT  CallModule, LoaderResults;
FROM    ShellMsg        IMPORT  SrcPaths, TextName, ErrorMsg, TextCol,
                                TextLine, ScanMode, CodeName;

CONST   EditorStackSize     =       0L;
        CompilerStackSize   =       0L;
        DriverStackSize         =     10000L;

VAR     ErrMsg                                  : ARRAY[0..79] OF CHAR;
        LoaderResult                            : LoaderResults;
        ExitCode                                : INTEGER;
        DriverModInstalled, quit                : BOOLEAN;
        key                                     : CHAR;

PROCEDURE LoadDriver ( DriverName : ARRAY OF CHAR; VAR Installed: BOOLEAN );
BEGIN
  Writeln; WriteString ( 'Loading driver "' ); WriteString ( DriverName );
  WriteString ( '" ' );
  IF ~Installed THEN
    CallModule ( DriverName, StdPaths(), '', DriverStackSize,
                 ExitCode, ErrMsg, LoaderResult );
    IF LoaderResult # NoError THEN
      CASE LoaderResult OF
      denied:      WriteString ( '... resident module.' )|
      notremoved: WriteString ( '... not removed.' )|
      notFound:    WriteString ( '... driver not found.' )|
      ELSE WriteString ( '... error.' ); END; Read ( key );
    ELSE WriteString ( '... loading ok.' ); Installed := TRUE END;
    IF ExitCode # 0 THEN WriteString ( '/unkown exitcode.');
      Read ( key ); END;
  ELSE
    WriteString ( ' => is installed.'); Read ( key );
  END; Writeln;
END LoadDriver;

PROCEDURE ReloadDriver ( Name: ARRAY OF CHAR; VAR Installed: BOOLEAN );
BEGIN
  Writeln; WriteString ( 'Reloading driver "' );
  WriteString ( Name ); WriteString ( '" ' );
  IF Installed THEN
    CallModule ( Name, StdPaths(), '', DriverStackSize,
                 ExitCode, ErrMsg, LoaderResult );
    IF LoaderResult # NoError THEN
      CASE LoaderResult OF
      denied:      WriteString ( '... resident module.' )|
      notremoved: WriteString ( '... not removed.' )|
      notFound:    WriteString ( '... driver not found.' )|
      ELSE WriteString ( '... error.' ); END;
      Read ( key );
    ELSE WriteString ( '... reloading ok.' ); Installed := FALSE; END;
  ELSE WriteString ( ' => was not installed.'); END;
  Writeln;
END ReloadDriver;

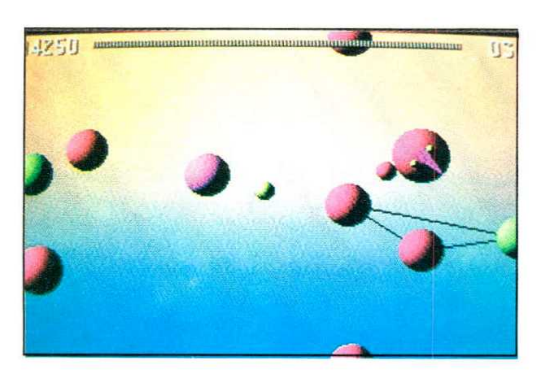
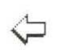
PROCEDURE EditDriver;
VAR     CompErr, sok, ok, NewLink : BOOLEAN;
        arg, Name                  : ARRAY[0..79] OF CHAR;
BEGIN
  Name := 'C:\Driver.TXT'; (* Das Textfile zum Treiber. *)
  CompErr := FALSE;
  REPEAT
    ok := FALSE; NewLink := FALSE; WritePg; ExitCode := 0; arg := Name;
    IF CompErr THEN
     (* Angabe der Fehlerposition im Textfile des Treibers: *)
      Append (' ',arg,sok); Append (CardToStr (TextLine,0),arg,sok);
      Append (' ',arg,sok); Append (CardToStr (TextCol-1,0),arg,sok);
      Append (' "',arg,sok); Append (ErrorMsg,arg,sok); Append ('"',arg,sok)
    END;
    CallModule ( 'GEP_ED', StdPaths(), arg,
                 EditorStackSize, ExitCode, DriverErrMsg, LoaderResult );
    CompErr := FALSE; WritePg;
    IF LoaderResult # NoError THEN
      WriteString ( ErrMsg ); Writeln; Read ( key ); ok := TRUE;
    ELSE
      CASE ExitCode OF
      1, (* Verlassen des Editors mit 'C',dann nur neu compilieren. *)
      2: (* Verlassen des Editors mit 'X',dann compilieren und neu linken.*)
        IF ExitCode = 2 THEN NewLink := TRUE; END;
        ScanMode := FALSE; ExitCode := 0;
        arg := Name; Append (' ',arg,sok); Append ('Q+',arg,sok);
        CallModule ( 'M2', StdPaths(), arg,
                     CompilerStackSize, ExitCode, DriverErrMsg, LoaderResult );
        WritePg;
        IF LoaderResult # NoError THEN
          WriteString ( DriverErrMsg ); Writeln; Read ( Key );
          NewLink := FALSE; ok := TRUE;
        ELSE ok := (ExitCode = 0); CompErr := ~ok; END;
      ELSE ok := TRUE; END;
    END;
  UNTIL ok;
  IF NewLink THEN
    ReloadDriver ( 'Driver', DriverModInstalled );
    LoadDriver ( 'Driver', DriverModInstalled );
  END;
END EditDriver;

BEGIN
  LoadDriver ( 'Driver', DriverModInstalled );
  REPEAT
    WritePg; Writeln; WriteString ( '  [1] edit driver' ); Writeln;
    WriteString ( '  [q] quit' ); Writeln; Read ( key );
    CASE CAP ( key ) OF
    '1': EditDriver|
    'Q': quit := TRUE|
    END;
  UNTIL quit;
END LinkTrick.
```

Abb.5: Ein skizziertes Megamax-Modula-2-Programm zur Verkürzung der Turn-Around-Zeit. Erläuterung siehe Text.

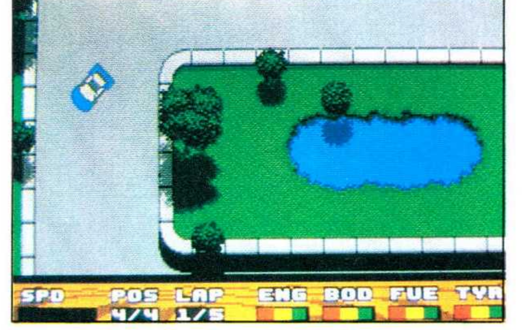
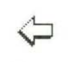


Bei "E-Motion" geht es darum, mit Bindfäden verbundene Kugeln mit Hilfe eines Raumschiffs zusammenzuschubsen. Kollidieren zwei gleichfarbige Kugeln, verschwinden sie vom Bildschirm und erst wenn alle Kugeln weg sind, geht's weiter in den nächsten Level. Die Schubserei findet natürlich unter Zeitdruck statt. Wer einen Mitspieler findet, kann das Game auch im Simultanmodus genießen. Gelungene Grafik und ausgezeichnete Spielbarkeit machen "E-Motion" zu einem Suchtspiel.

E-Motion

"Super Cars": Noch nie war ein Autorennspiel so spannend. Um sich für's nächste Rennen zu qualifizieren, muß man unter den ersten drei Rennwagen sein, die durchs Ziel rauschen. Spannung und Dramatik sind vorprogrammiert. Schließlich geht's um hohe Preisgelder, mit denen man sein Rennauto besser ausstatten oder gar ein neues kaufen kann. Die Startampel springt von gelb auf grün, Vollgas! Bloß nicht beim Überholen gegen die Seitenbegrenzung knallen - das ist für die Reifen nicht besonders förderlich. Ebenso sind Crashs zu vermeiden. Die Schadensanzeige am unteren Bildschirm zeigt warum: Sinkt sie auf Null, ist das Spiel vorbei und das Preisgeld futsch. Die Rennen sind trotz Darstellung aus der Vogelperspektive unheimlich packend.

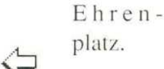
Super Cars

"Stryx", das neueste Spiel aus der englischen Software-Schmiede "Psygnosis", beeindruckt anfangs sehr stark. Der Vorspann besitzt exellente Grafiken und Sounds, wie wir es von "Psygnosis" gewohnt sind. Doch beim eigentlichen Spiel scheint den Programmierern die Puste ausgegangen zu sein: ein Plattformspiel allerprimitivster Sorte. Statt der Grafiken vom gewohnten Roger Dean-Standard erscheint alles in so kunterbunten Farbtönen, daß einem fast die Brillengläser zerspringen. Auch die Animationen sind ein ganz schlechter Scherz. Der Held bewegt sich wie ein rheumatischer Orang-Utan. Aber damit nicht genug: Noch träger hätte die Joystick-Steuerung gar nicht sein können, von der mangelhaften Spielbarkeit einmal ganz zu schweigen.




"Stryx"

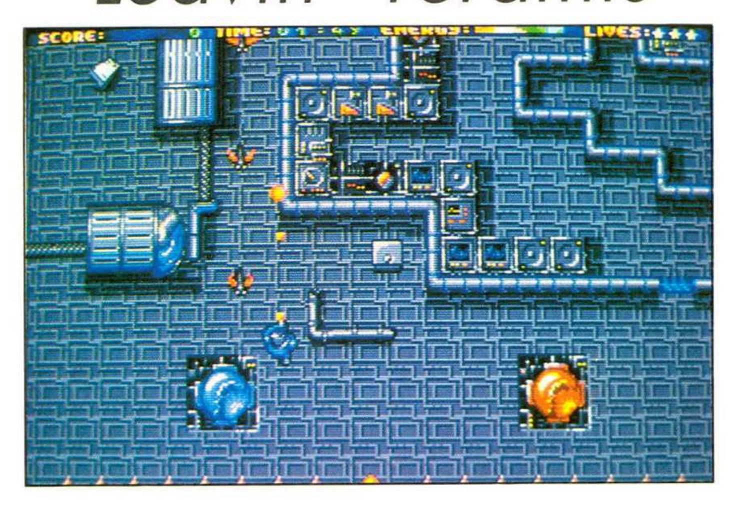
"Pipe Mania" besitzt Hitqualitäten: Ein einfaches, aber süchtig machendes Spielprinzip. Es geht schlicht und ergreifend darum, auf einem 7*10 Kästchen großen Feld aus vorgegebenen Einzelteilen eine Pipeline zusammenzubasteln. Man spielt es allein oder zu zweit im Simultanmodus. "Pipe Mania" hat mir beim Testen viel Freude gemacht. Grafik und Sound sind zwar nur überdurchschnittlich, jedoch geht vom Spielprinzip eine ungeheure Faszination aus. Schnelle Reaktionen sind genauso gefragt wie Kombinations- und Konzentrationsvermögen. Empires neues Tüftelspiel ist ein Superhit und verdient in jeder Software-Sammlung einen Ehrenplatz.





Pipe Mania

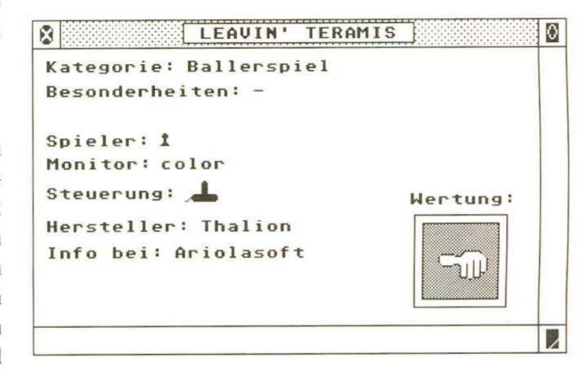
Leavin' Teramis



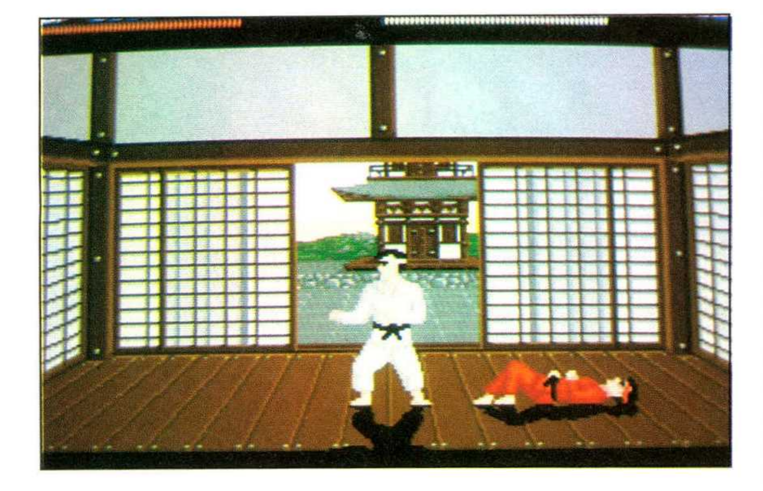
Die Party gestern war nicht schlecht. Eigentlich war sie sogar gut. Für Nigel MacGibbons war sie, genauer gesagt, etwas zu gut. Nigel, Hyperdrive-Ingenieur dritter Klasse an Bord des ehrenwerten Forschungsraumschiffes Teramis, hat nämlich auf der Feier in den xenobiologischen Labors ungefähr die Menge reinen Alkohols zu sich genommen, die ein xyrellanischer Hanggleiter in zwei Stunden im Leerlauf verbraucht (wieviel das auch immer sein mag). Und als ob die heute in seinem Kopf umhermarschierende Zwergenarmee noch nicht Strafe genug wäre, haben sich während seines Ausnüchterungsschlafes auch noch a) alle Monster aus der Abteilung für außerirdische Lebensformen befreit und b) alle Besatzungsmitglieder der Teramis daraufhin mit den Rettungskapseln aus dem Staub gemacht. Im Rausgehen haben sie dann schnell noch die Schalter für Licht und Selbstzerstörungsanlage gedreht. So findet sich MacGibbons jetzt in seiner selbstgewählten Ausnüchterungszelle am Heck des Schiffes, durch ein ganzes Schiff voller Monster getrennt von der einzigen verbleibenden Rettungskapsel im Bugraum. Doch wie sagte schon der Lastwagenfahrer vor der viel zu niedrigen Unterführung so richtig: "Da muß man halt durch...". Und genau dabei soll der Spieler Nigel MacGibbons helfen. Das Resultat: Ein Ballerspiel, das weder sich selbst noch den Rest der Welt ernstnimmt.

Die einzelnen Abteilungen der Teramis (=Spiel-Level) strotzen nur so von Monstern, die sich die Thalion-Jungs von anderen Quellen "entliehen" haben. Besonders bei den Endgegnern der Level finden sich viele alte Bekannte (kennen Sie Audrey II, die nette fleischfressende Pflanze von nebenan?). So ganz nebenbei muß der Spieler noch gegen die Uhr kämpfen, denn die Selbstzerstörungsanlage der Teramis läuft unerbittlich weiter. In den Seitenarmen des Raumschiffs trifft man auf weitere starke Gegner, mit denen man sich aber trotzdem anlegen sollte: Die wichtigsten Extrawaffen, Ausrüstungsgegenstände und manchmal sogar etwas Extrazeit findet man hier. Ab und zu allerdings verbirgt sich hinter den fiesesten Monstern auch schon mal eine Niete... Technik und Ideen können sich also sehen lassen, auch der Sound geht ganz gut ab. Nur - fanatische Ballerspielfreunde mögen es mir verzeihen - was soll's? Auf Dauer mangelt es dem Ganzen an Spielreiz. So lustig auch die Einfälle sind, irgendwann fehlt einfach die Motivation, Teramis nochmal zu laden. Zudem ist das Spiel stellenweise so schwer, daß ohne die Continue-Funktion schnell erste Frustbißspuren an der Diskette zu entdecken wären. Aber, die oben erwähnten Ballerspielfreunde jetzt bitte aufgepaßt, es ist zu schaffen. Gerüchte besagen, daß Programmierer und Spiel-Designer Erik Simon sein Spiel schon mal bis zu Ende geschafft haben soll. Ohne Continue. Ob er es aber ein zweites Mal versuchen würde, wage ich zu bezweifeln...

CBO



Chinese Karate



"Turtle Byte" hat mit "Chinese Karate" ein neues Prügelspiel auf der Pfanne. Ein oder zwei Spieler liefern sich Tritte in die Magenkuhle, Handkantenschläge in den Nacken oder Fausthiebe ins Gesicht. Spielt man alleine gegen einen Computerkarateka, gilt es, acht Runden bestehend aus zwei Kämpfen ohne K.O. zu überstehen. Sieger eines Kampfes ist, wer nach 90 Sekunden noch über die meiste Lebensenergie verfügt. Nach jeder Runde folgt ein Zwischenspiel, in dem Sie herumpurzelnde Bälle, herumfliegende Drachen oder Tonkrüge zerschlagen müssen. Dann beginnt die nächste Runde mit neuer Hintergrundgrafik und einem stärkeren Gegner. Es lohnt sich übrigens, alle Gegner zu besiegen: Knallharten Karatekämpfern wird von einer hübschen Geisha eine goldene Medaille verliehen. Wenn das kein Anreiz ist? Da wir gerade von Goldmedaille sprechen: Eine solche würde ich sofort dem Sound-Programmierer verleihen: Chinamucke mit fetzigen Schlagzeugeffekten - da freut sich das Ohr! Auch der Grafiker verdient Bewunderung: Saubere Animationen und teilweise echt beeindruckende Hintergrundgrafiken verstehen zu entzücken.

Spielerisch geht "Chinese Karate" auch in Ordnung. Der Schwierigkeitsgrad steigert sich von Runde zu Runde angenehm. Der einzige herbe Kritikpunkt ist die Tatsache, daß man mit dem Ellenbogenschlag den Gegner fast immer von den Füßen fegt.

CBO



Skizzenbücher von Volkmar Hoppe entdeckt! 11

Wer mit Text- und DTP-Programmen arbeitet, braucht hochwertige und aussagekräftige Grafiken zur copyright-freien Verwendung: Die Skizzenbücher von Volkmar Hoppe enthalten tausende von bisher unveröffentlichten Zeichnungen, Objekten und typografischen Elementen. In hoher Auflösung gescannt und sorgfältig nachbearbeitet, stellt diese Grafikserie einen aussergewöhnlichen Fundus an Ideen und vielseitigen Clip-Arts bereit, mit dem Sie über Monate hinweg erfolgreich gestalten und publizieren. Lassen Sie sich inspirieren: Sie werden staunen.



Hoppe 1 **DM 179,-** *

120 Seiten DIN A4 auf 15 Disketten. Alle Dateien im GEM Image-Format: In Calamus, 1st Word und ähnlichen Programmen direkt verwendbar. Handbuch und Konvertierungshilfe.

CADjA V.1.2 **DM 998,-** * 10

CAD-Programm der Spitzenklasse, anwendbar für Elektronik, Maschinenbau, Architektur. Plotausgabe bis DIN A0. Symbolbibliotheken verfügbar. ASCII- Schnittstelle. Direkte Weiterverarbeitung mit "CADjA-CAM". Schnittstelle zu STEVE. Leistungsmerkmale: überragend! Druckausgabe auf Nadeldrucker, Laser, Plotter. Demo: DM 50,- *

STar Designer **DM 149,-** *
ARABESQUE **DM 278,-** *

Love & Fun 13

zeigt, welcher Partner zu Ihnen paßt: In der Liebe, beim Sport und am Arbeitsplatz. Kern der Profi-Version ist die schnelle Adressverwaltung, deren gesamter Datenbestand zur Partnersuche herangezogen werden kann. Mächtige Analysemöglichkeiten, flexible Suchkriterien, einfache Bedienung und die elegante Oberfläche zeichnen Love&Fun aus. **DM 59,-** *

Dr. Lustig **DM 29,-** *
Der Psychoanalytiker. Ein toller Partyspaß.

11	
CALAMUS 1.09	**DM 799,-** *
OUTLINE ART	**DM 398,-** *
Reprostudio ST	**DM 498,-** *
Retouche	**DM 399,-** *
PKS-Write	**DM 198,-** *
CALAMUS Fonteditor	**DM 198,-** *
DIDOT Fonteditor	**DM 198,-** *
Layout-Paket für Calamus	**DM 79,-** *

Gestaltungshilfe, Pass- und Schnittmarken in DIN Standartformaten A5, A4 und A3, hoch u. quer. Mit Anleitung (auf Wunsch in Englisch).

ST-DCL **DM 298,-** * 14

Kommandozeileninterpreter + Tools ST-DCL erlaubt es, auf einem Atari ST unter einer Kommandoumgebung zu arbeiten, die der VAX/VMS-Digital Command Language (DCL) nachempfunden ist. Auch für MS-DOS lieferbar.

ST Pascal plus **DM 249,-** *
SPC Modula 2 **DM 348,-** *
STAN **DM 69,-** *
Zur statistischen Auswertung und grafischen Darstellung von Zahlen und Messreihen.

Architektur & Design **DM 59,-** * 11

Über 400 Vektor- und Rastergrafiken (GEM-/PAC-Format) für CALAMUS.

Geben Sie Ihrem Calamus die Fonts, die er braucht!

Plup, Bobble, Skript, ROCA, YAPPIE, Caslo, ROMA REG, AltBerlin bold, CARDPLAY, Peking reg, Rund, JILLY, INFRA, AKTIVA, GEODET, Boedet

Desk Assist 4.2 **DM 188,-** * 19

Das Multifunktions-Accessory! Terminplaner mit Alarm- und Dauerterminen, Adressen- und Telefondatei für Serienbriefe, Druckerspooler, Taschenrechner, Maßumrechnung, RAM und Diskeditor, ASCII-Tabelle und noch mehr ...

SIM Rechner-Freezer **DM 35,-** *
Speichert den aktuellen Betriebszustand des ST auf Diskette ab. "Eingefrorene" Programme können schnell und einfach reaktiviert werden.

Quick-Screen **DM 49,-** *
Beschleunigt Bildschirmaufbau u. Textausgabe.

Layout-Paket für Calamus **DM 79,-** * 11

Gestaltungshilfe, Pass- und Schnittmarken in DIN Standartformaten A5, A4 und A3, hoch u. quer. Mit Anleitung (auf Wunsch in Englisch).

Marconi Trackerball **DM 198,-** *
100%tiger Mausersatz mit hoher Lebensdauer.

Handyscanner Typ 10 **DM 898,-** *
400 dpi, 105mm Scannbreite, Texterkennung und Bildverarbeitungsprogramm.

Scan-Paket **DM 3598,-** *
Scanface ST + Panasonic FX-RS 505 mit 400 dpi und Pseudo-Graustufen im Ditherverfahren.

20

5.25"	Laufw., 40/80 Trks.	**DM 318,-** *
3.5" NEC	Laufwerk	**DM 259,-** *

Slim-Line, anschlußfertig und durchgeschliffen.

EL 3.5"	Laufwerk	**DM 248,-** *
EX 30L	Festplatte	**DM 1298,-** *
EX 60L	Festplatte	**DM 1998,-** *
EX 40W	Wechselplatte	**DM 2698,-** *
MEGA-DRIVE 40A		**DM 1398,-** *

Einbauplatte 40MB für alle MEGA ST.

ST-Uhr 2 für 520/1040ST **DM 79,-** *
Auf Soundchip steckbar, einfacher Einbau.

EM 124 Multi-Monitor (f & s/w) **DM 598,-** *

VOMBLATT **DM 89,-** * 12

Musikdidaktisches Programm zum Erlernen des Notenlesens - auch ohne Vorkenntnisse - bis Hochschulniveau. Alle Schlüssel, Ein-/Ausgabe über MIDI möglich, Prüffunktionen und mehr...

Steinberg 'Twelve' **DM 99,-** *
12 Spur Midi Sequencer. 'Twelve' ist der kleine Bruder des schon populär gewordenen Steinberg Twenty-Four.

Cubase **DM 790,-** *
Desktop Midi Recording System

RAM-Erweiterung 2,5 MB **DM 898,-** * 20

Mit 4 MB Speicherkarte (steckbar) von Weide.

RAM-Erweiterung 4 MB **DM 1498,-** *
Wie oben, jedoch mit vollen 4 MB bestückt.

A1MB Ram-Erweiterung **DM 548,-** *
A3MB Ram-Erweiterung **DM 1148,-** *
Neu: Der bereits vorhandene Speicher Ihres ST bleibt einsatzbereit, Sie benötigen somit nur A3MB zur Aufrüstung von 1 MB auf 4 MB.

MiniRAM 1MB-Erweiterung **DM 298,-** *
4-Bit organisierte Erweiterung für 260/520 ST.

RR - Soft
Grundstraße 63
5600 Wuppertal 22
0202 / 640389

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt 90
069 / 763409

IDL Software
Lagerstraße 11
6100 Darmstadt 13
06151 / 58912

KREATIV-Software
Oberwürzbacher Str. 10
6676 Mandelbachtal
06803/3850 u. 06805/2666

Musikinstrumente&Computer
August-Bebel-Str. 3
6840 Lampertheim 5
06241 / 80899

Weeske Computer
Potsdamer Ring 10
7150 Backnang
07191 / 1528-29 od. 60076

Duffner's PD-Center
Ritterstr. 6
7833 Endingen a.K.
07642 / 3875 od. 3739

T.S. Service
Szemere Hard & Software
Schleißheimerstr. 127, 8 Mü 40
089 / 3089408

Dietmar Schramm
Promberg 6
8122 Penzberg
08856 / 7287

Lipstick

Schick EDV-Systeme
Hauptstraße0 32a
8542 Roth
09171 / 5058-59

PR8-SOFT, Klaus-M. Pracht
Otto-Hahn-Str. 10
8702 Estenfeld
09305 / 8211

Bossart-Soft
Sonnenhofstr. 25
CH-6020 Emmenbrücke
041 / 534182

* Alle Preise sind unverbindlich empfohlene Verkaufspreise

Coupon

Ich bestelle:

Name, Vorname

Straße

Plz, Ort STC 6/90

☐ Per Scheck. Zuzüglich DM 5,- Versandkosten, unabhängig von der bestellten Stückzahl.
☐ Per NN (Nur Inland, zuzügl. DM 8,- NN-Gebühr)

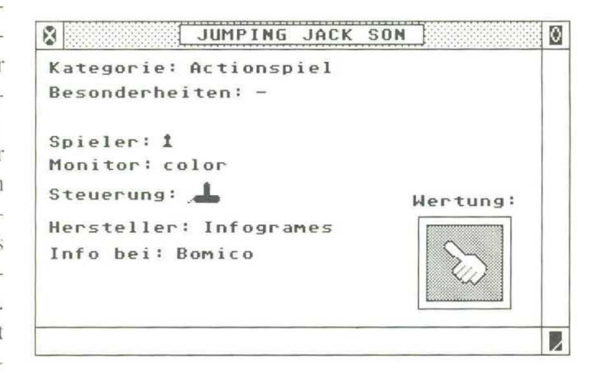
Jumping Jack Son



Der Rock 'n Roll ist tot, es lebe die klassische Musik! Überall machen sich Serenaden, Konzertinos und Etüden breit, trampeln Baßgeigen und Pauken auf den Gehörnerven der Rockfans herum. Diese Zustände sind natürlich für den Spieler untragbar, und so macht er sich in dem neuesten Spiel von Infogrames auf den Weg, den Rock zu retten. Als Hauptfigur steht ihm dabei der Junior von Jumping Jack Flash zur Seite, der sich, gesteuert per Joystick, durch diverse Level kämpft, um die guten alten Rockscheiben wiederzufinden. Diese Level bestehen aus unterschiedlich angeordneten Feldern, auf denen Klein Jack - nomen est omen - herumjumpt, was die Sprunggelenke hergeben. Einzelne Bereiche aus farbigen Feldern lassen sich durch Überqueren umfärben. Gelingt es Jack, einen kompletten Bereich einheitlich zu kolorieren, erscheint eine Schallplatte, die er aufnehmen und zu einem leeren Plattenspieler der gleichen Farbe bringen muß. Drauf mit der Scheibe, und schon erklingt das erste Instrument in rassigem Rhythmus! Natürlich sollte Jack darauf achten, die Bereiche auch in den Farben anzulegen, für die es noch freie Plattenspieler gibt. Als kleines Hilfsmittel zeigen farbige Balken unten auch an, welche Farben noch gefragt sind. Tönt von allen Plattenspielern eines Levels feinster Rock, kann Jack sich mit einem Sprung auf ein Schachbrettfeld in den nächsten Level katapultieren. Klingt soweit doch ganz einfach, oder?

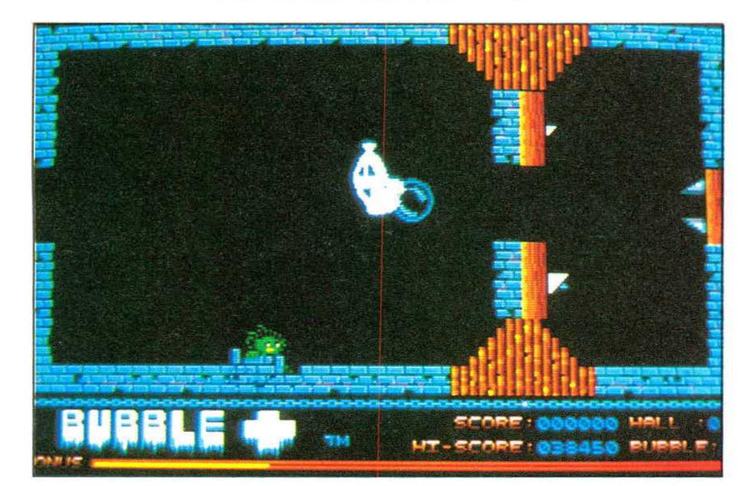
Wäre es auch, wenn da nicht die bösen Gegner wären: Bösartige Baßcellos, tödliche Trommeln und gefährliche Geigen machen Jack das Leben schwer. Sie rasen über das Spielfeld und versuchen, ihm eines seiner drei Leben auszuhauchen. Zur Gegenwehr stehen ihm Rockkassetten zur Verfügung, die den Gegner eine Weile aufhalten, Schlaftabletten, um feindliche Instrumente für kurze Zeit medikamentös ruhigzustellen und Sonnenbrillen, die ihm ein derart furchterregendes Aussehen geben, daß alle Gegner kreischend die Flucht ergreifen. Hamburger, Pizzascheiben und Cola-Flaschen werden unterwegs gern mitgenommen, das gibt Extrapunkte, eine schnittige Lederjacke steht für ein zusätzliches Leben. Sammelt Jack einen Walkman auf, wird er entweder unsichtbar, unbesiegbar oder besonders schnell. Der Spieler verfolgt den ganzen Musikkrieg aus der Vogelperspektive. Er steuert den herumhüpfenden Irrwisch und legt mit der Feuertaste die Tape-Barrikaden aus. Alle vier Levels kommt eine Challenge-Runde auf ihn zu, in der er jedes der Felder mit Jack nur einmal betreten darf. Gelingt es ihm, alle Felder in der vorgegebenen Zeit abzuklappern, erhält er ein Password, mit dem er direkt vom Start in höhere Level einsteigen kann (versucht doch mal, als Password "Elvis" einzugeben, und viel Spaß in Level 13...). Das Ganze macht technisch einen sauberen Eindruck, die musikalische Untermalung ist mit gesampelten Rockausschnitten gewürzt, wird aber dennoch schnell zur Ohrfolter. An der Grafik gibt es nichts zu meckern, und auch die Steuerung von Jack Junior ist einfach und präzise. Wer sich weit genug emporgehüpft hat, wird namentlich in die Top 1000 (!) aufgenommen.

CBO


JUMPING JACK SON

Kategorie: Actionspiel
Besonderheiten: -
Spieler: 1
Monitor: color
Steuerung:
Hersteller: Infogrames
Info bei: Bomico
Wertung:

Bubble +



Infogrames neues Geschicklichkeitsspiel "Bubble +" ist schlicht und ergreifend enttäuschend. Das putzige Geisterbild auf der Verpackung kann nicht darüber hinwegtäuschen, daß es sich hierbei um ein erbärmliches Game handelt, das nicht nur durch grafische Lieblosigkeit, sondern auch durch fehlenden Spielwitz unangenehm auffällt. Der Spieler hat nichts weiter zu tun, als einen Gespenster-Sprite per Maus über den Bildschirm zu steuern. Der Geist kann Drehungen entweder nach links oder nach rechts vollführen, je nachdem, welche Maustaste man drückt. Ziel des Spielchens ist es nun, durch Drücken der Space-Taste eine Blase zum Ausgang zu pusten. Dazu muß das Gespenst immer in der richtigen Position zur Blase stehen, damit sie ihren Weg nimmt und nicht an Hindernissen hängenbleibt - das ist schon alles, was der Spieler tun kann. Grafisch ist "Bubble +" sehr primitiv gehalten: die einzelnen Spielabschnitte bestehen aus schlicht zusammengesetzten Steinchen und hölzernen Bauteilen, die stark an Spielgrafiken zur Pionierzeit des ATARI ST erinnern. Was den Sound angeht, wird auch nicht viel mehr geboten: Es gibt eine Titelmelodie und simple Geräuscheffekte während des Spielverlaufs. Wenn Sie mich fragen, machen Sie im Laden einfach einen großen Bogen um "Bubble +", um sich eine Enttäuschung zu sparen.

CBO

BUBBLE +

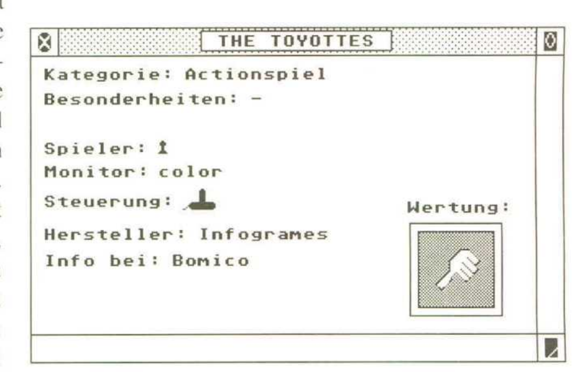
Kategorie: Geschicklichkeitsspiel
Besonderheiten: -
Spieler: 1
Monitor: color
Steuerung:
Hersteller: Infogrames
Info bei: Bomico
Wertung:

The Toyottes



Aus Angst vor bitterbösen Monstern haben sich riesige Rattenrudel in eine unterirdische Welt geflüchtet. Dort herrscht ein tyrannischer König, dessen einziger Sohn Barnabas heißt. Eines schönen Tages verschwindet der kleine Bubi spurlos. Als der König das Verschwinden seines heißgeliebten Stammhalters bemerkt, schickt er eine Ratte auf die Suche. Kommt sie ohne den Königssohn zurück, will man ihr jedes Haar einzeln aus dem Fell ziehen. Soviel zur Hintergrundgeschichte von Infogrames neuem Comic-Spielchen "The Toyottes". Warum der französische Hersteller das Programm in seine Comic-Serie eingeordnet hat, ist mir völlig schleierhaft, eine entsprechende Grafik ist nämlich kaum vorhanden. Die mickrige Ratte ist nur durch ein paar Pixel-Punkte dargestellt. Aber nicht nur die Grafik ist etwas dürftig ausgefallen. Auch das Spielprinzip lockt im Jahre 1990 keinen ST-Spieler hinterm Ofen hervor. Man läuft mit der Ratte durch die Abenteuerwelt, klettert Leitern und Seile empor, fährt mit Aufzügen und rutscht durch riesige Rohre, um von einem Ort zum anderen zu gelangen. Außerdem sammelt man noch gelegentlich Gegenstände ein und futtert Käsestückchen. Zwischendurch prügelt sich die Ratte mit Monstern, wie zum Beispiel mit dem Wächter. Dieses Biest wird zahm, wenn man ihm einen bestimmten Gegenstand mitbringt. Dann gibt es noch Verfolger, die erst Ruhe geben, wenn sie dem Helden Lebensenergie entzogen haben. Die Pillis futtern nur den Käse weg, und dann gibt es noch eine andere Monsterart, die Rohrleitungen ausbeult. Ich muß sagen, ich habe mich beim Testen fürchterlich gelangweilt. Neben "Bubble +" schon der zweite Flop in diesem Monat aus dem Hause Infogrames. Die französische Company scheint momentan ein kleines Formtief zu haben.

CBO



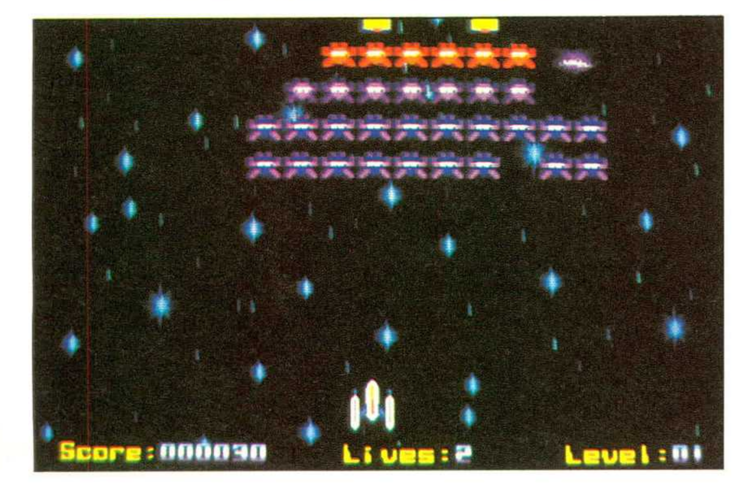
THE TOYOTTES

Kategorie: Actionspiel
Besonderheiten: -

Spieler: 1
Monitor: color
Steuerung:
Hersteller: Infogrames
Info bei: Bomico

Wertung:

Renaissance



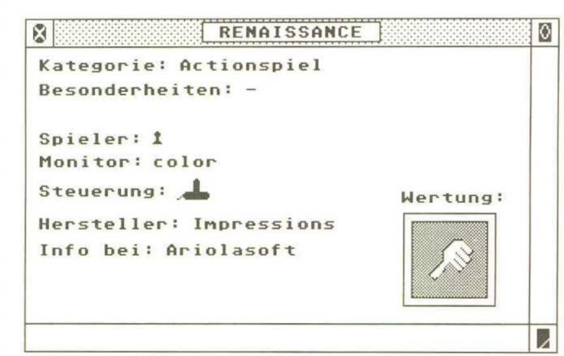
Das englische Label "Impressions" ermöglicht einen tiefen Griff in die Oldie-Kiste: Space Invaders, Asteroids, Centipede und Galaxions - alles Spiele aus den Anfangszeiten der Homecomputer.

Neben den klassischen Versionen gibt es auf der Diskette noch aufgemotzte Varianten mit etwas besserer Grafik und neuen Spielelementen.

Für diejenigen unter Ihnen, die Invaders, Asteroids, Centipide und Galaxions noch nie gesehen haben, möchte ich sie noch einmal kurz vorstellen: In "Space Invaders" steuert man ein Raumschiff auf einer horizontalen Achse. Ein schießender Alien-Schwarm rückt immer um eine Zeile tiefer, droht langsam die Barrikaden zu zerstören und auf der Achse zu landen. Damit dies nicht geschieht, muß der Spieler die immer schneller werdenden Alien-Schiffe abschießen. Bei "Asteroids" steuert man ebenfalls ein Raumschiff über den Bildschirm. Heranfliegende Asteroiden müssen kleingeschossen werden, damit die Gesteinsbrokken nicht mehr das Schiff treffen. In "Galaxions" steuert man genau wie bei "Invaders" ein Raumschiff auf einer horizontalen Achse und ballert angreifende Aliens ab. "Centipide" führt in einen Garten, in dem man mit einem kleinen Geschütz auf Hundertfüßler und allerlei anderes Getier ballert. Die Spielprinzipien und die technische Umsetzung aller vier Klassiker sind sehr simpel und können mit dem heutigen Standard nicht mithalten.

"Renaissance" läßt sich nicht mit der gewohnten Qualität des Jahres 1990 vergleichen, man muß dieses Produkt eher als Sammelobjekt begreifen. Deshalb könnten wir bei der Bewertung von Grafik und Sound eigentlich sehr nachsichtig sein. Das gilt zumindest für die klassischen Spielversionen. Doch leider besitzen selbst die aufgemotzten Fassungen eine sehr klägliche Sound- und Grafikqualität. Fazit: Nur für unerschrockene Sammler!

CBO



RENAISSANCE

Kategorie: Actionspiel
Besonderheiten: -

Spieler: 1
Monitor: color
Steuerung:
Hersteller: Impressions
Info bei: Ariolasoft

Wertung:

WOBO-Service
Willi B. Werk

Signum! Zwei	a.A.
STAD 1.3+	151,--
Script	171,--
Headline 3.0	86,--
That's Write Profi	298,--
Calamus	691,--
GFA-BASIC 3.5	216,--
GFA-ASSEMBLER	119,--
GFA-STRUKTO	206,--
GFA-DRAFT-plus	305,--
ChemGraf	73,--
TEMPUS V.2.0x	102,--
Diskus	125,--
1ST-ADDRESS	89,--
NeoDesk	76,--
Hard Disk Sentry	116,--
Btx/Vtx-Man. o.l.	245,--
Anti Viren Kit 3.0	79,--
Adimens ST plus	329,--
AdiTALK ST plus	329,--
THEMADAT 4.0	209,--
TiM I	119,--
BS-Fibu/3	689,--
fibuMAN e	329,--
Harlekin	112,--
Mortimer	79,--
CISYTEM	108,--
SciGraph	539,--
Lavadraw	138,--
DRAW! 3.0	129,--
MegaPaint II	396,--
Arabesque	216,--
PC-Speed V.1.4	439,--
SpeedBridge	68,--
Supercharger 1MB	715,--
Fuji MF2DD 10St.	28,--

Alle Preise in DM, inkl. der gesetzl. MWSt.
Vorbehaltlich Irrtümer und Preisänderungen.
Bei Vorkasse 2% Skonto, zuzügl. DM 5,50 Versandkostenanteil; bei Nachnahme kein Skonto, zuzügl. DM 9,50 Versandkostenanteil. Kein Ladenverkauf!
Dies ist nur ein kleiner Ausschnitt aus unserer Preisliste, die wir Ihnen gerne kostenlos übersenden.

Sielwall 87, D-2800 Bremen 1 - Tel. 0421/75116

ATARI

Mega ST 1 / SM 124	DM	1528,–
Mega ST 2 / SM 124	DM	2128,–
Mega ST 4 / SM 124	DM	3098,–
Megafile 30	DM	848,–
Megafile 44	DM	1998,–
Laser SLM 804	DM	2348,–
+ Postscript	+ DM	298,–

Multitasking - Betriebssystem für ATARI - ST auf Anfrage

Alle Preise incl. Mwst.. Wir führen nur deutsche Originalware direkt vom autorisierten ATARI-Händler und geben 1 Jahr (!) Garantie.

KRÜGER EDV - MARKETING
Tel. 0 28 57 / 17 01 Fax. 0 28 57 / 17 00
Rees ☆ Düsseldorf ☆ Viersen

KK Computer-Service Kohler
Don - Carlos - Str. 33 B, 7000 Stuttgart 80



29,00 DM
(zuzügl. 3,00 Porto)

Monochrommonitor und doppelseitiges Laufwerk werden benötigt.

EXPEDITION NACH VARTORIA...

Expedition nach Vartoria ist ein klassisches Rollenspiel, ein Abenteuer und eine Herausforderung für jeden, der es wagt sich darauf einzulassen!

Sie werden Städte erkunden mit Piraten kämpfen, durch den Dschungel marschieren und vieles mehr.

Und Sie werden tatsächlich einige Geheimnisse aufdekken, aber wahrscheinlich nicht alle...

Public-Domain Software
Die PD-Disks aus den Serien:
! "ST-COMPUTER" und "2000er Serie" .... DM 5,00
! Eigene Sammlung (England, USA und Deutschland) .... DM 4,50
! ABO der Serien...3,90 ! PD-Pakete.....20,00 (zuzügl. 5,00 Porto)
Wir machen PD-Software wieder günstig!
BITTE GRATISKATALOG ANFORDEN!!!

Computer & Electronic & Zubehör HERGES
Obere Rischbachstraße 88 • 6670 St. Ingbert
Telefon (06894) 38 31 78 / Telefax (06894) 38 28 55

Atari-Computer + Zubehör:
Mega-St's, Maus, Blitter, Basic, Handbuch, Anschlußkabel usw.

ST2 mit 2 MB RAM DM 1998,- *ST2 mit 4 MB RAM aufger.	DM	2658,-
Atari Mega-ST mit PC-Speed MS-DOS-Emu eingebaut, Aufpreis	DM	500,-
Megafile 30, kpl. Anschlußkabel, Software und Handbuch	DM	888,-
Megafile 44, incl. 1*Cartridge (Zubehör wie Megafile 30)	DM	2128,-
Wechselcartridge (SQ 400/44 MB) für Megafile 44	DM	268,-
Laserdrucker SLM-804, Software, Anschlußkabel, Interface	DM	2698,-
Toner-Cartridge-Kit für Laserdrucker SLM-804, komplett	DM	95,-

Atari Ersatz-Chips:

Soundchip	25,-	Glue 147	147,-	MFP	29,-	Shifter	168,-
WDc 1772-2	39,-	MMU	147,-	DMA	147,-	Blitter	188,-
Tast.Proz.	69,-	68000	21,-	6850	4,-	RP5-C15	28,-
TL-497ACN	8,35	TL-7705	3,90	PC900	4,65	74LS/usw a.A.	

Dyn. RAMs + Gals + Gal-Zubehör:

DRAM 1 MB*1, Typ 51.1000-70 ns (andere xxx-ns a.A.)	DM	23,45
DRAM 256 KB*1, Typ 41.256 100 ns (andere xxx-ns a.A.)	DM	7,40
2 MB RAM-Erweiterung (0 MB bestückt) auf max. 2.5/ 4 MB	DM	128,-
2 MB RAM-Erweiterung (2 MB bestückt) auf max. 2.5/ 4 MB	DM	540,-
Gal 16V8-25L (Sockel für Gal 16-V8-25L DM -,35)	DM	4,90
Gal 20V8-25L (Sockel für Gal 20-V8-25L DM -,40)	DM	6,50
Gal-Programmer für Gals 16V8 und 20V8, incl. Software, ab	DM	130,-
Gal-Assembler für die Erstellung der Fuse Matrix	DM	148,-

Eproms + Eprommer + Platinen:

E-Prom 27C256-200 ns (32 KB*8) Programmiersp. 12.5 V.	DM	7,90
E-Prom 27C512-200 ns (64 KB*8) Programmiersp. 12.5 V.	DM	13,70
E-Prom 27C010-200 ns (128 KB*8) Programmiersp. 12.5 V.	DM	31,60
Epromdisk 512 KB kompl. mit Software, 0 KB bestückt	DM	158,-
Epromdisk-Erweiterung (auf max. 1 MB) 0 KB bestückt	DM	88,-
ST-Eprommer mit Software, bis Eproms 27011/27C011	DM	168,-

IC-Sockel LC-Low Cost / PZ-Präz. / SM=schmal:
* 64pol. PZ=6.75 * 48pol. PZ=2.95 * 40pol. LC=0.75 * 40pol. PZ=2.25
* 32pol. LC=0.55 * 28pol. LC=0.45 * 24pol. LC=0.42 * 24pol. SM=0.40
* 20pol. LC=0.35 * 18pol. LC=0.30 * 16pol. LC=0.25 * 14pol. LC=0.20
* 08pol. LC=0.15 * Einzelkontakt-PZ=0.07 - Stecker-PZ=0.10

Sonstiges aller Art:
Schaltnetzteil-Platine in Industriequalität.....!!!
Kompl. aufgebaut + geprüft, kurzschlußfest, superleicht, dyn. Stromentnahme bis 60 W, VDE 0804/0806 TÜV Rheinland, Abm. 160x100x45 mm,

Typ + 5V/5 Amp., + 12 V/2.5 Amp., -12 V/0.5 Amp.	DM	67,-
Star LC-24/10, 7 KB-Puffer, Paper Park-Funktion	DM	728,-
Quarz-Oszillator 10.24 MHz oder 16 MHz, bitte angeben	DM	8,-
PC-Speed (MS-DOS auf dem Atari ST)	DM	118,-
Mr.Robot Actionspiel mit 18 Levels	DM	39,-

Auszug aus unserem Lieferprogramm, bitte Komplettliste anfordern!
Lieferung zuzügl. Porto/Verp. per Post! Alle Angebote freibleibend!
***** Bei Terminvereinbarung auch Selbstabholung *****

SOFTWARE • IMPULSE Atari ST
COMPUTER-SERVICE KOHLER
Don-Carlos-Str. 33B, 7000 Stuttgart 80

Bitte Gratiskatalog *"Software - Impulse"* anfordern!

VIDEO STAR: neues, megastrakes Videoverwaltungs- und Bearbeitungsprogramm. *Video Star* hat neben den "normalen" Verwaltungsarbeiten, wie z. B. Ausdruck von Listen, Etiketten oder Diskettenoperationen (Formatieren, Ordner, Fremdprog.) einen besonders ausführlichen Filmbeschreibungsteil, wie u. a. Schauspieler, Trick, Drehbuch, Filmmusik, Regie, Filmart, Spieldauer, Kamera, Aufgenommen am oder Kopie von (dies sind nicht alle Funktionen). Zum Schluß gelangen Sie in einen kl. Textteil, in dem Sie noch eine Inhaltsangabe verewigen können............DM 49,--*

HASCS - THE GAME CREATOR: ein Rollenspielentwicklungsset. Zum Erstellen eigener Fantasyspiele benötigen Sie keine Programmierkenntnisse, sondern nur eine gute Fantasie. Die Spiele "Allein in Eritra" oder "Rat der Weisen" wurden auch mit *HASCS - THE GAME CREATOR* erstellt.......................DM 59,--*

HASCS - EXPEDITION NACH VARTORIA: dieses fesselnde Spiel wurde mit HASCS - THE GAME CREATOR erstellt. *Expedition nach Vartoria* ist ein Abenteuerspiel für Fantasyfreunde mit guten Nerven, Kombinationsfähigheit und viel Zeit.. Kämpfen Sie gegen Piraten, marschieren Sie durch den Dschungel für..DM 29,--*

SIEGFRIED HÜBNER's LISTING PRINTER: ist ein kleines aber sehr nützliches Druckerutility, das speziell zum Ausdruck von Listings oder langen Texten entwickelt worden ist. *Listing Printer* bringt noch so lange Texte in gestochen scharfer Minischrift zu Papier!.......................DM 15,--*

PUBLIC DOMAIN DISKETTEN: Wir führen auch alle PD-Disketten der 2 großen PD-Serien. Eine Disk kostet bei uns nur ....DM 5,--*
Disks im ABO nur DM 3,90. PD-Pakete (5 doppels. Disks) nur DM 20,--

*zuzügl. DM 5,-- (Porto) / Originalsoftware nur für hohe Auflösung, 2DD

PORTRAITS UND MODEGRAPHIKEN

für DTP und Graphikanwender. Verschiedenste Einsatzmöglichkeiten, zur Werbung, Ausgestaltung von Layouts. Eigens entworfene Graphiken, gescannt mit 200 bis 400 dpi. Nicht auf Screenformat begrenzt. Je nach Bildgröße auf mehrere Screens verteilt; inclusive jeweils ein Gesamtbild. Somit auch für 600x400 Monitore nutzbar. Sehr gute Bildqualität beim Drucken. Auch bei Orginalgröße. Nutzbar auch für Anwender die nicht über Vektorgraphik verfügen (Signum!). Über 800k gepact auf Markendisketten. Einführungspreis : 60.-- DM incl. Porto und Verpackung. Zusätzlich Unterlagen zur Zusammensetzung der einzelnen Bilder. Verrechnungscheck, bar oder Überweisung. Zur Bestellung genügt eine Postkarte. Auf Wunsch fertigen wir auch spezielle Graphiken an. Weitere Grapiken in Vorbereitung.



LUKIDIS G. - KIRCHHEIMER STR. 54-1 - 7440 NÜRTINGEN
Bankverbindung: Volksbank Nürtingen Konto-Nr: 27 31 BLZ: 612 901 20

> ATARI ST <

Anwendersoftware

CCD

ST Pascal + V2.0x	220,-
Tempus Editor 2.0	100,-
Tempus Word	a.A.
Assembler Tutorial	90,-

GFA

GFA Basic 3.0 (I.+C.)	180,-
GFA Assembler	140,-

Omikron

Omikron Basic Comp.	170,-
Mortimer, Utility	75,-

Appl. Syst.

Signum!2	418,-
Scarabus	90,-
Signum Revers Acc.	90,-
Protos	64,-

Fontdisketten verfügbar

Spielesoftware

F-16 Combat Pilot	69,-
Gunship	79,-
Indiana Jones (Adv.)	69,-
Oil Imperium	53,-
Populous	69,-
Rick Dangerous	69,-
RVF Honda	69,-
Sleeping Gods Lie	69,-
Spherical	53,-
Zak McKraken	69,-
Star Trash	50,-
Diskbox 3,5" 80er	19,-
PC-Speed	a.A.
Traktrix	80,-

Approximationsprg. für sämtl. Fkttypen

Porto: Vorkasse 4,- Nachnahme 7,- DM

Computerversand G. Thobe
Pf 1303 - 4570 Quakenbrück
Tel.: (05431) 5251

Elektronik Software

professionelle Elektronik - Simulatoren für alle Atari ST

ASPICE
Industriell einsetzbarer Transistor-Analogsimulator mit bester GEM-Grafik für Atari ST

Optional: Halbleiter-Modellbibliothek, Spezialversion auch für 68000/68881 und 68020/68881

Preis zuzüglich Porto und Verpackung **ab DM 348.-**

ADIGSI neu
Industriell einsetzbarer Worst-Case Digitalsimulator mit Schaltungseingabe für Atari ST

Optional: 74LSxx-Bauteile-Bibliothek, Spezialversion auch für 68000/68881 und 68020/68881

Preis zuzüglich Porto und Verpackung **ab DM 348.-**

erfordert Joystickport für Kopierschutz
alle Atari ST ab 1MByte RAM

CAE SYSTEME RUFF Dipl.-Ing.
Tel. 07307-24187 FAX -23209
D-7910 Neu-Ulm Pf. 1942

HARDWARE für Ihren Atari ST

Speichererweiterungen inklusive Einbau

260/520ST	auf 1,0 MB	198.-
520ST+/1040ST	auf 2,0 MB	548.-
1040STE	auf 2,5/4 MB	498.-/948.-
Mega 1	auf 2/4 MB	548.-/1048.-
Mega 1 mit 2MB	auf 4 MB	548.-
Mega 2	auf 4 MB	ab 548.-

Preise für Platine/Einbausatz bitte anfragen

Lüfter und Sonstiges:

Leiser Lüfter in Mega 1/2/4/Harddisk	je 59.-
Monitorbild SM124 um 25% größer	39.-
Floppy-Umschaltung (A/B, Side 0/1)	je 49.-
TOS-Umschaltung	49.-
Umbau 2«-»6 EPROMs	89.-

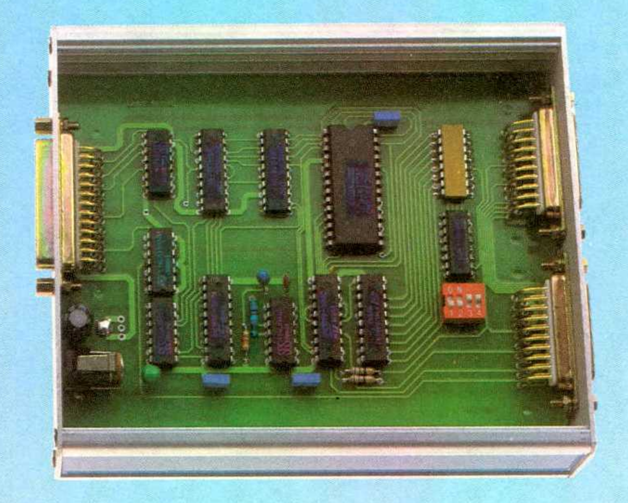
Weitere Umbauten, Sockelservice a. Anfrage

xxxxxxxxxxxxxxxx Info gratis xxxxxxxxxxxxxxxx

FME Computerservice
Dipl.-Ing. Fritz Metternich, Mozartstraße 7
Tel. ab 17h: (0 61 83) 21 77, 6455 Erlensee









Das UNIVERSALINTERFACE für Panasonic-Scanner am ATARI ST (inkl. Scansoft)

Unterstützt alle 3 Panasonic-Scanner.
Automatische Scanner-Erkennung

	898,– DM
Versandkosten	6,– DM
Gesamtpreis	904,– DM

Überzeugend: DIE HARDWARE

Unterstützt alle 3 Panasonic Scanner
Modelle: FX-RS505 (max. 400 dpi, Pseudograustufen durch Ditherverfahren)
FX-RS506 (max. 400 dpi, 16 echte Graustufen)
FX-RS307 (max. 600 dpi, 256 echte Graustufen)

Scannen und Speichern aller gängigen Bildformate möglich:

IMG	(GEM-Image Format, für z.B. Calamus usw.)
TIF	(Standard-TIFF Format = ermöglicht Datenaustausch mit IBM-kompatiblen PC's)
IFF	(Amiga-Standard Format)
BLD	(Megapaint Format)
PIC, PAC	(Stad Format)
PI3	(Degas Format)
IMC	(Signum u. Creator Format)

Druckertreiber:
Für folgende Drucker sind entsprechende Treiber vorhanden:
HP-Laserjet kompatible
Canon Laserdrucker
ATARI Laserdrucker
NEC P6 und kompatible 24-Nadeldrucker
EPSON kompatible 9-Nadeldrucker

Begeisternd: DIE SOFTWARE

- Arbeitet mit allen gängigen Festplatten zusammen: z.B. ATARI Megafile 30/60/4, VORTEX 20/30/60
- Scannen von Schwarz-Weiß und Graubildern
- Mehrere Bilder in versch. Fenstern gleichzeitig bearbeiten
- Schnelles Scrollen durch die Bilder
- Maßangaben alternativ in Pixel, cm oder Zoll
- Kopieren, Verkleinern, Vergrößern von Bildteilen möglich, aber zwischen verschiedenen Bildern (bei echten Graubildern ohne Qualitätsverlust)
- Wandlung von Schwarz-Weiß nach Graubild und umgekehrt.
- Verbesserte Bildausgabe durch Laserwert, d.h. der Laser kann beeinflußt werden, den Hintergrund heller oder dunkler zu machen (nur bei Panasonic Laserdruckern)
- Auslagerung von zu großen Graubildern auf die Festplatte
- Nachbearbeitungen von allen Bildern mit dem Editor möglich, d.h. Pixel (einzelne Bildpunkte) können gesetzt und gelöscht werden
- Lupenfunktion = 4-fache Vergrößerung, dadurch einfache Nachbearbeitung
- Ausschnitte löschen oder kopieren, d.h. es kann aus mehreren einzelnen Bildern ein neues Bild erstellt werden.
- Grauwertscala (Histogramm), d.h. anhand der Grauwertscala ist zu erkennen, in welchem Grauwertbereich man sich befindet.
- Lineare Solarisation, d.h. Helligkeit (Kontrast) können verändert werden.
- Aquidensiten, d.h. die Anzahl der Graustufen kann verringert werden (Konturenverbesserung).
- Deutsches Handbuch

Überragend: DAS BESONDERE

1. Das Zusammenarbeiten mit allen gängigen Festplatten
2. ***Scanface*** ermöglicht Bildbearbeitung und Ausgabe schon mit kleinem Rechnerspeicher (520 ST). Bereits beim 520 ST kann der ATARI-Laserdrucker eingesetzt werden.
3. Aufgrund der guten und strukturierten Programmierung der Software sind folgende weitere besonderen Vorteile gegeben:
   — Software benötigt kaum Arbeitsspeicher
   — ATARI-Laserdrucker benötigt keinen eigenen Druckerpuffer
   — Extrem schnelle Umrechnung von Bildern von **Schwarz-Weiß nach Grau** oder von **Grau nach Schwarz-Weiß**
   — Unterschiedlichste Rasterverfahren möglich z.B. ● Floyd-Steinberg ● Dither ● 5 direkt anwählbare Raster (auch bei Panasonic FX-RS 505 zu benutzen)
4. Automatische Erkennung des installierten Scanner-Modells
5. Konsequentes Ausnutzen der Scanner-Möglichkeiten.

bei Ihrem Fachhändler oder beim Heim-Verlag

BESTELLCOUPON

_____ ***Scanface*** Universalinterface für Panasonic-Scanner am ATARI ST à 898,– DM
(Unverbindlich empfohlener Verkaufspreis)
Zahlung ☐ per Scheck ☐ per Nachnahme zuzügl. Versandkosten DM 6,– (Ausland DM 10,–)
unabhängig von der bestellten Stückzahl

Name: ______________________

Straße: ______________________

PLZ, Ort: ______________________

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Internationaler Vertrieb:
Compo Software GmbH
Tel.: 0 65 51 / 62 66

Schweiz
Data Trade AG
Landstr. 1
CH - 5415 Rieden - Baden

Österreich
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

Wordplus-Texte in TeX-Qualität

Die hervorragende Ausgabequalität von TeX dürfte sich mittlerweile herumgesprochen haben: Buchsatzqualität - und das Public Domain. Da der ATARI ST ja gerne als bessere Schreibmaschine benutzt wird, mag so manche Studentin und so mancher Autor mit dem Gedanken spielen, sein altgedientes Wordplus durch TeX zu ersetzen.

Die Sache hat aber einen Haken, nämlich die Bedienung von TeX. Nix Maus und nix WYSIWYG. Kryptische Kommandos sind angesagt. Statt Fettschrift, die auch dem Bildschirm als solche erkennbar ist, verlangt TeX ein trockenes *\bf*. Und sind dann die wichtigsten Steuerbefehle gelernt, fängt der Streß erst an. Wie bringe ich das Programm dazu, ausgerechnet hier oder dort keinen Seitenumbruch zu machen und das Wort "Urinstinkt" nicht nach der zweiten Silbe zu trennen?

```
Fettschrift wird in Fettschrift übersetzt (\bf)
Unterstreichen wird in Unterstreichen übersetzt (\underbar{})
Kursiv wird in Kursivschrift übersetzt (\it)
Hell wird in Fixpunktschrift übersetzt (\tt)
superscript wird in superscript übersetzt ($^{ }$)
subscript wird in subscript übersetzt ($_{ }$)
```

Gerade die Leute, die am meisten schreiben, möchten sich in der Regel auf ihre Textarbeit konzentrieren und sich nicht mit den für andere Leute sicher spannenden Geheimnissen der Maschine herumschlagen. Deshalb haben sie sich ja den ATARI ausgesucht, wo alles so schön anschaulich zu bedienen ist. Zum Rechner gesellt sich dann gewöhnlich noch ein Drucker und Wordplus: ein durchaus typischer Arbeitsplatz für Leute, die ihren Computer fast ausschließlich für Textverarbeitung brauchen. Tatsächlich muß man dem guten alten Wordplus bei allen Schwächen doch zugute halten, daß es die wichtigsten Anforderungen des täglichen Schreibens abdeckt. Was die Einfachheit der Bedienung betrifft, dürfte es wohl unübertroffen sein. Die Funktionen beschränken sich auf das Notwendige und sind auch von Anfängern schnell zu erlernen. Der Griff zum Handbuch ist kaum erforderlich. In meinen Augen ist das nicht das schlechteste Konzept.

Aber schön wär's natürlich doch, wenn denn wenigstens die Examensarbeit oder Dissertation, das selbstverlegte Buch oder andere größere Werke besser gedruckt werden könnten. Aus dieser Situation heraus ist *WPlusTeX* entstanden (PD-Disk 299). Das Programm konvertiert Texte, die mit Wordplus geschrieben wurden, in TeX-Quelltexte... und plötzlich erscheinen sie tatsächlich in TeX-Qualität.

Doch ganz so einfach, wie das jetzt klingt, geht's eben doch nicht. Wer jetzt glaubt, ein TeX mit der Bedienungsoberfläche von Wordplus vor sich zu haben, wird enttäuscht. Nach der Konvertierung sind gewöhnlich noch einige Nachbereitungen erforderlich. Da wird die Seite an einer ungünstigen Stelle umgebrochen, bestimmte Einrückungen erscheinen nicht so, wie sie gedacht waren, usw. Hier muß sich nach wie vor ein TeXperte oder auch eine TeXpertin dransetzen und manuell nachbessern. Die Idee ist eher folgende: Leute, die eben ihre Texte schreiben und ansonsten nicht viel mit Computern zu tun haben wollen - eine durchaus verständliche Einstellung -, erstellen, ganz wie gehabt, ihre Texte unter Wordplus und wenden sich dann, wenn der Text inhaltlich steht, an jemand in ihrem Bekanntenkreis mit TeX-Kenntnissen. (Für eine Einladung zum Essen oder durch sonstiges gutes Zureden lassen die sich gewöhnlich überreden.) Dort wird der Text einmal durch das Konvertierungsprogramm WPlusTeX gedreht und kommt als TeXdatei heraus. Die Arbeit des/der TeXkundigen beschränkt sich dann darauf, das äußere Layout den Wünschen entsprechend zu gestalten, ein paar Nachbesserungen vorzunehmen und schließlich den entgültigen Ausdruck zu besorgen.

Die wichtigsten Merkmale des Textes werden vollautomatisch übertragen. Fett- und Kursivschrift, Unterstreichungen, hoch- und tiefgestellte Passagen werden entsprechend umgesetzt; Einrückungen, Zentrierungen und rechtsbündige Zeilen erscheinen auch so in TeX. Fußnoten und Grafiken werden ebenfalls übersetzt. Unkomplizierte Texte können oft sogar ohne nennenswerte Korrekturen in TeX ausgedruckt werden.

Einige Dinge lassen sich allerdings nicht sinnvoll übertragen. Tabellen beispielsweise müssen völlig neu angeordnet werden; woher sollte das Programm schließlich wissen, daß irgendwelche Wörter mit Absicht genau bündig untereinander stehen? Der Textinhalt der Tabelle braucht immerhin schon nicht mehr neu getippt zu werden. Über komplexe mathematische Formeln im Text braucht wohl kein Wort gesagt zu werden, aber H_2O und x^2 klappen allemal. Wenn Sie Texte neben Grafiken anordnen, werden auch Nacharbeiten im Layout nötig, das Kopieren der Grafikdatei, der TeX-Befehl zum Einbinden der Grafik und die horizontale Plazierung der Grafik werden Ihnen aber bereits abgenommen.

Wenn das Programm den betreffenden Autoren/innen auch keinen direkten Zugang zu den Leistungen TeXs ermöglicht, scheint es mir für größere Texte doch eine nützliche Brücke zu schlagen.

Heinz-Günther Fischer

Dieser Text ist für alle diejenigen gedacht, die scl
gearbeitet haben und nun TEX lernen wollen.

TEX (sprich Tech, wie in Technik) ist kein Textvera
mit der Schreibmaschine treten.

TEX ist ein Textsatz-System d. h. TEX will in Konkı
strikt in Hinblick auf optimale Druckergebnisse entw
ist ergo nicht vorgesehen. Dies hat einige Konsequeı

- Die Druckertreiber können nicht an den Zeich *schrift*, **Fettschrift** oder vergrößern werden dern für jede Kombination der „Effekte" mu vorhanden sein.

  Daher redet man bei TEX auch nicht von „Efi Familien.

- Die fonts werden nicht von Hand für ein spez in parametrisierter Form vor und sind leicht Ausgabegerät zu produzierten. Diesen Zweck ‹

  Sie können also Ihre Ausgabe ohne Probleme : Linotype Satzanlage belichten lassen. Sie erhal

```
Dieser  Text ist für alle diejer
mit einem Textverarbeitungsprogr
lernen wollen.

TeX  (sprich Tech,  wie in Techr
system d.  h. TeX will nicht in
ne treten.

TeX  ist ein Textsatz-System d.
Buchsatz· treten und wurde infol
optimale Druckergebnisse entwick
aussehen  könnte,  ist  ergo nic
Konsequenzen:

    - Die  Druckertreiber  könne
      (fonts)  manipulieren.  Ef
      schrift oder vergrößern we
      treibern  realisiert,  son
      "Effekte" muß (für jedes A
      satz vorhanden sein.
```

Zum Vergleich: Links Probeausdruck mit TeX und rechts Probeausdruck mit 1st Wordplus

Folgende TeX-Disketten sind bis jetzt in unserem PD-Service erschienen:

TeX-Grundpaket	**PD 250 & 251**
P6-Druckertreiber	**PD 252**
PD-TeX-Metafont	**PD 253 & 254**
Grafikfähige Druckertreiber	**PD 276**
Laserdrucker-Fonts	**PD 277**
8/9-Nadeldrucker-Fonts	**PD 278**
Makros und Fonts (Musik, Chemie)	**PD 284**
WPTeX	**PD 299**

GAL-Prommer

Junior Prommer





MGP - Einfach, schnell und praktisch

Die Zeiten der elektronischen Schaltungen, als Röhren mit ihrer Heizwendel oder gar Relais viel Strom und Platz verbrauchten, sind längst vorbei. Moderne Computerbausteine und Logik-Chips finden sich in fast jeder Schaltung wieder. Kein Fernseher, keine Aufzugssteuerung und schon gar keine Computerschaltung kommt ohne Steuerlogik und Verknüpfungsbausteine aus.

Aber auch die Zeiten, in denen man mit unzähligen sogenannter TTL-Gräber oder CMOS-Bausteinen eine Schaltung entwickelte, gehen vorüber. Nach den frei programmierbaren, aber nicht löschbaren PAL-Bausteinen werden immer häufiger die frei programmierbaren und löschbaren und somit viel flexibleren GAL-Bausteine eingesetzt. Kein Wunder, denn durch ein einziges GAL lassen sich viele der herkömmlichen Logikbausteine ersetzen.

Neben der enormen Platz- und Stromersparnis bleibt eine einmal aufgebaute Schaltung flexibel, denn ein GAL ist in wenigen Sekunden umprogrammiert und kann somit neuen Anforderungen angepaßt werden. Typische Einsatzgebiete eines GALs sind:

1. **Adreßdekoder**
2. **Zustandsautomaten**
3. **logische Gatter**
4. **PAL-Simulation**

Der MAXON GAL- Prommer 16/20 bietet Ihnen alle Möglichkeiten, die bekannten GAL-Typen 16V8 und 20V8 mühelos zu programmieren. Nutzen Sie diese modernen Bausteine, um Ihre Schaltungen schnell, preiswert und flexibel aufzubauen. Mit dem MGP 16/20 kein Problem, denn im ausführlichen Bedienungshandbuch wird auch erklärt, wie herkömmliche Logik (NOR-, NAND-, NOT-, ...-Gatter) in ein GAL programmiert wird und wie diese Bausteine arbeiten.

Der MGP 16/20 läuft an jedem ATARI ST und wird an die Druckerschnittstelle angeschlossen. Die benötigte Betriebsspannung wird am Joystick-Port abgenommen. Die menügesteuerte Software ermöglicht ein bequemes und sicheres Arbeiten mit den GALs. Im eingebauten Editor läßt sich die JEDEC-Datei (so heißt die Datei, die in ein GAL programmiert wird) leicht erstellen oder ändern.

Ein Muß für jeden Elektroniker - GALs, die neue Generation der programmierbaren Logik!

Klein, kompakt und leistungsstark

Der Junior Prommer programmiert alle gängigen EPROM-Typen (und deren CMOS-Typen), angefangen vom 2716 (2 kByte) bis zum modernen 27011 (1 MBit). Mit dem optionalen Mega-Modul-Adaptersockel lassen sich sogar die neuesten 32-pol. EPROM-Typen 27010, 27020, 27040 und 27080 brennen. Aber nicht nur EPROMs, sondern auch einige ROM- und EEPROM-Typen lassen sich lesen bzw. programmieren. Zum Betrieb benötigt der Junior Prommer nur +5 Volt, die am Joystick-Port Ihres ATARI ST abgenommen werden, alle anderen Spannungen erzeugt die Elektronik des Junior Prommers. Selbstverständlich läßt sich ein 16-Bit Word in ein High- und Low-Byte zerlegen. Fünf Programmieralgorithmen sorgen bei jedem EPROM-Typ für hohe Datensicherheit. Im eingebauten Hex/ASCII-Monitor läßt sich der Inhalt eines EPROMs blitzschnell durchsuchen oder auch ändern.
Bemerkenswert ist der Lieferumfang, so wird z.B. das Fertiggerät komplett aufgebaut und geprüft im Gehäuse mit allen Kabeln anschlußfertig geliefert. Auf der Diskette mit der Treibersoftware befinden sich noch RAM-Disk und ein Programm zum Erstellen von EPROM-Karten, ferner wird der Source-Code für Lese- bzw. Programmierroutinen mitgeliefert und last but not least ist im Bedienungshandbuch der Schaltplan abgedruckt.

Typ

Int.ID,			
EPROM'S			
2716	25V	2732	25V
2732A	21V	2764	21V
2764A	12V	27128	21V
27128A	12V	27256	12V
27256	21V	27512	12V
27513	12V	27011	12V
Mit MEGA-Adapter			
✓ 27010	12V	27020	12V
27040	12V	27080	12V
ROM'S			
4732		4764	
47128		47256	
EEPROM'S			
X2804A		X2816A	
X2864A		X28256A	



Übersicht der mit dem Junior Prommer programmierbaren Speichertypen

Profiline

macht den ROM-Port zum vielseitigsten Port des ATARI ST

Was ist das Profiline-System?

Mit dem Profiline-System können Sie den ROM-Port (auch Modul-Port genannt) nach Belieben erweitern. Dazu stehen verschiedene Karten zur Verfügung, die je nach Bedarf ausgebaut werden können.

Der Profitreiber

Es handelt es sich dabei um eine Treiberkarte, die direkt in den ROM-Port eingesteckt wird, und alle Adreß-, Daten- und Signalleitungen verstärkt, so daß ein problemloser Betrieb aller weiteren Karten an jedem ATARI ST-Modell gewährleistet ist. Ferner ermöglicht diese Karte auch den Schreibzugriff am ROM-Port.

Die Profibank

Die Profibank besteht aus einer EPROM-Bank und einem frei programmierbaren Eingabe-/Ausgabe-Port.
Die EPROM-Bank kann bis zu 12 EPROMs der Typen 27512 oder 27011 aufnehmen, so daß maximal 1,5 Megabyte ROM-Speicher zur Verfügung stehen. Programme, Daten, Accessories und Autostart-Programme lassen sich auf diese Weise sicher speichern.
Sie benötigen zum Programmieren einen Eprom-Brennner (z.B. Junior Prommer).
Der Eingabe-/Ausgabe-Port stellt dem Anwender 32 frei programmierbare Leitungen und 4 Kontrollleitungen (flankenempfindlich) zur Verfügung. Mit diesem E/A-Port lassen sich beliebige Steuerungsaufgaben erledigen.

Das Profi-RAM

Alleinstehend oder auch optional zur Profibank stellt das Profi-RAM voll bestückt bis zu 384 kByte akku-gepufferten RAM-Speicher zur Verfügung. Durch das Profiram, das höchste Priorität besitzt, kann jederzeit bestimmt werden, ob vom Profiline-System gebootet werden soll oder nicht. Natürlich ist auch ein Schalter vorhanden, um das RAM vor ungewolltem Überschreiben zu schützen.
Zum Betrieb des Profi-RAMs bzw. der Profibank ist der Profitreiber nötig!

Alle Platinen sind fertig aufgebaut und geprüft!

Easytizer

Der Videodigitizer ohne Geheimnisse

Mit dem Easytizer können Sie beliebige Videosignale von einer Schwarzweiß- oder Farb-Kamera, Videorecorder oder direkt vom Fernsehgerät (mit Composite Video Ausgang) digitalisieren und somit auf dem Bildschirm Ihres ATARI ST sichtbar machen.

Der Easytizer wird am Modul-Port des ST angeschlossen. Die Auflösung beträgt 800x600 Bildpunkte, so daß in mittlerer Auflösung 640x200 Bildpunkte in vier Graustufen dargestellt werden können. In dieser Betriebsart werden 12,5 Bilder in der Sekunde wiedergegeben. Im hochaufgelösten Modus werden 640x400 Bildpunkte in Schwarzweiß wiedergegeben.

Besondere Merkmale des Easytizers:

- Software vollständig in Assembler
- Abspeichern der Bilder im DEGAS-Format
- Von STAD und Sympatic Paint ansteuerbar
- Ein beliebiger Bildausschnitt kann in ein mit der Maus wählbares Format vergrößert und verkleinert werden
- Spiegeln eines Bildes in horizontaler und vertikaler Ebene
- Animation mit beliebig vielen Bildern möglich, nur durch die Kapazität des Rechners begrenzt (beim Mega ST4 über 100 Bilder)
- Eingebauter Druckertreiber für die mittlere Auflösung für NEC P6/P7 und EPSON oder Kompatible
- Wahlweise automatische oder manuelle Helligkeitseinstellung
- Schnappschuß

Lieferumfang:

1) Fertiggerät: komplett aufgebaut und geprüft, inclusive Diskette mit der Easytizer Software und Bedienungsanleitung

2) Teilsatz: Doppelseitiger, elektronisch geprüfter Platine mit Lötstoplack und Bestückungskungsaufdruck sowie vergoldeten Anschlußkontakten, fertig programmiertes GAL 16V8, Quarzoszillatormodul 32 MHz, Diskette und Bedienungsanleitung.



Bestellcoupon MAXON Computer GmbH Schwalbacherstraße 52 6236 Eschborn Tel.: 06196/481811

Name: ____________________

Vorname: ____________________

Straße: ____________________

Ort: ____________________

Unterschrift: ____________________

☐ Vorauskasse
☐ Nachnahme

Versandkosten:

Inland	DM	7,50
Ausland	DM	10,00

Auslandsbestellungen **nur** gegen Vorauskasse
Nachnahme zuzgl. DM 4,00 Nachnahmegebühr.

Hiermit bestelle ich:

☐ Profitreiber (wird f. Profibank und Profi-RAM benötigt)	DM	129,00
☐ Profibank (inkl. Profiport)	DM	125,00
☐ Profi-RAM ohne stat. RAMs	DM	119,00
☐ Gehäuse z. Einbau v. Profibank u. Profi-RAM	DM	39,00
☐ MGP-GAL-Programmiergerät	DM	195,00
☐ MGP-Platine, -Software und -Gehäuse	DM	95,00
☐ Junior Prommer ST (Fertiggerät wie beschrieben)	DM	229,00
☐ Adaptersockel Mega Modul f. Junior Prommer	DM	99,00
☐ Junior Prommer Leerplatine und Software (o. Bauteile)	DM	59,00
☐ Junior Prommer Leergehäuse (gebohrt und bedruckt)	DM	39,90
☐ Junior Prommer AMIGA Fertiggerät	DM	249,-
☐ Junior Prommer MS-DOS Fertiggerät inkl. Netzteil	DM	348,-
ROM-Karte 128 kByte bietet maximal 4 EPROMs Platz ☐ (fertigbestückt o. EPROMs)	DM	58,00
☐ Easytizer (Fertiggerät)	DM	289,00
☐ Easytizer (Teilsatz wie oben beschrieben)	DM	129,00

Vertrieb in der Schweiz: DTZ DataTrade AG Landstrasse 1 CH-5415 Rieden-Baden Tel.: 056/821880 Fax.: 056/821884
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3
Vertrieb in Frankreich: AROBACE 2Rue Piemontesis F-75018 Paris Tel.: 1/42235044 Fax 1/42545631

FPU-Precompiler
für Modula-2

Teil 1

Unser geliebter ATARI wird immer mehr für "nützliche" Dingen eingesetzt. Im DTP-Bereich hat er sich längst seinen Platz erobert und auch die Scharen derer, die mit selbstgeschriebenen Programmen ihren Rechner für Stunden und Tage in das Reich der Mathematik schicken, wächst zusehens. Daher ist es an der Zeit dem Computer etwas nachzuhelfen, damit dem auch so bleibt, denn schließlich hat nicht jeder einen Mainframe am Netz.

Das Problem und seine Lösung

Je mehr man seinen ST nicht nur zum Spielen und Texteschreiben benutzt, sondern z.B. auch für naturwissenschaftliche Problemstellungen, desto mehr Rechnerzeit wird für die Berechnung mathematischer Ausdrücke benötigt. Auch wenn man "nur" ein paar Fraktale berechnet werden sollen, wird schnell der Wunsch nach einem schnelleren Rechner laut.

Integrale, Fourier-Transformationen und endlose Reihen komplexer Zahlen erfordern eben viele Fließkommaoperationen und das ist genau das, was der Prozessor 68000 von Haus aus nicht kann. Er kann lediglich ganze Zahlen mit einer recht beschränkten Genauigkeit handhaben und muß mit dieser Fähigkeit die Fließkommaoperationen "von Hand" durchführen. Die Programmierung dieser Arithmetikroutinen ist ein hartes Stück Arbeit und es ist verständlich, daß von Compiler zu Compiler starke Abweichungen in puncto Genauigkeit und Geschwindigkeit auftreten. Daß diese mühsam erarbeiteten Prozeduren auch bei den besten Programmierern langsamer sind als eine in Silicium geätzte von Profis konzipierte Hardware-Lösung, ist verständlich.

Ein solcher Chip ist der 68881. Es gibt ihn in Versionen für 12, 16 und 20 MHz und ist als "rechte Hand" des 68020 gedacht. Da er recht eng mit dem Hauptprozessor zusammenarbeitet, nennt man solch einen Schaltkreis auch Coprozessor. Die Firma Motorola, die diesen Rechenknecht entwickelt hat, baut natürlich auch noch andere Prozessoren und daher wurde der 68881 so konzipiert, daß er ganz normal in den Adreßraum integriert werden kann, ohne gleich mehrere PALs brennen zu müssen. Im Prinzip läßt sich damit auch ein ZX 81 aus der "Frühgeschichte" des Homecomputerzeitalters problemlos tunen.

Ein weiterer Grund zur Freude ist, daß man diesen Chip auch im ATARI nutzen kann. Diese Freude wird allerdings dadurch getrübt, daß im Gegensatz zu den IBM-kombatiblen kein Sockel auf der Hauptplatine existiert, in den man das gute Stück stecken könnte. Dafür hat ATARI eine Erweiterungsplatine für die MEGA STs entwickelt und auch die Entwickler der MAXON Grafikerweiterung haben daran gedacht, dem Coprozessor einen Platz anzubieten. Wer keinen MEGA ST auf dem Schreib- bzw. Labortisch hat, kann sich von Zweitanbietern helfen lassen.

Nichts ist perfekt

Hat man das gute Stück auf einem dieser Wege in seinen Rechner integriert und wirft man den Compiler an - der hoffentlich den Coprozessor unterstützt - so wird man erst einmal enttäuscht sein. Von den mit der Kombination 68000/68881 theoretisch etwa erreichbaren 4µs für eine Addition oder Multiplikation wird man doch deutlich entfernt sein. Außerdem wird die Bibliothek der höheren Funktionen sicherlich recht lückenhaft, dafür aber in Assembler geschrieben sein, so daß man als Hochsprachenprogrammierer die Lücken auch nicht allzu leicht stopfen kann. Für diejenigen, die Umformungen wie $exp(x+iy) = exp\,x(cos\,y+i\,sin\,y)$ öfters in den Programmen einsetzen, ist es auch interessant, daß der Coprozessor Sinus und Cosinus gleichzeitig ausrechnen kann, dies aber wohl von keinem derzeit erhältlichen Compiler-Paket angeboten wird. Da der Coprozessor bisher eher ein Schattendasein geführt hat, kann man es den Entwicklern der Compiler auch nicht verdenken, daß sie sich anderen Problemen zuwandten.

Für einen Compiler würde es verständlicherweise einen riesigen Aufwand bedeuten, bis zu acht Variablen optimal im Coprozessor zu verwalten. Er müßte zunächst einmal die wichtigsten Variablen festlegen, das Nachladen und Zurückspeichern optimieren und beim Aufruf von Unterprogrammen auch noch den Kontextwechsel berücksichtigen. Der übliche Weg ist daher, die Variablen, z.B. die beiden Summanden, in die FPU zu schreiben, dort zu addieren und dann gleich wieder in den Speicher zu bringen. Da die meiste Zeit dabei mit dem Laden und Speichern von Variablen verbracht wird,

macht sich der Coprozessor nur wenig bemerkbar.

Im Vergleich zu einem Compiler hat es ein Mensch da schon einfacher. Er sieht meist auf den ersten Blick, welche Variablen ständig im Gebrauch sind und wo sich zeitintensive Schleifen befinden. Mit dem Precompiler kann man dies jetzt ausnutzen und explizit angeben, was in welches Register geladen werden soll und wie man die einzelnen Register miteinander verrechnen will. Oftmals lädt man am Anfang die Variablen, rechnet dann nur noch mit den Registern und holt sich am Schluß z.B. das fertige Integral aus der FPU. Um diese Informationen möglichst einfach in den Rechner zu bringen wurde dieser Precompiler geschrieben. Er vorübersetzt den Text, damit der eigentliche Compiler das gewünschte Programm erzeugen kann.

Wer in Assembler fit ist und diese "Sprache" liebt, ist natürlich fein 'raus, er sollte hier abbrechen und in [1] und [2] weiterschmökern. Er wird dann mittels der CPU-Register noch (aber nur etwas) schnellere Programme schreiben können, als bei Verwendung des vorliegenden Precompilers und anschließender Übersetzung. Der Arbeitsaufwand für solche Assembler-Programme sollte allerdings nicht unterschätzt werden.

Der Stand der Dinge

Das vorliegende Programm wurde mit dem Hänisch-Modula2 von Schwab Software geschrieben. Es unterstützt derzeit diesen Compiler, das PD-Modula der TU München, das TDI Modula, den SPC Compiler und das demnächst wohl erhältliche Update des Megamax Modula Systems. Die Versionen unterscheiden sich lediglich in der Art der produzierten Vorübersetzung und in der Importliste, nicht hingegen im eigentlichen Programmtext. Dadurch sollten alle mathematischen Routinen beliebig ausgetauscht werden können. Einer PD-Bibliothek für coprozessorgestützte Numerik steht also nichts im Wege.

Der Code wurde auf einen 16MHz-Coprozessor hin optimiert, schnellere Versionen sind daher nur bei höheren Funktionen sinnvoll. Wer einen langsameren Coprozessor hat, muß weitere Verzögerungen einfügen. Aber dazu später mehr.

Das Modulaprogramm, das vorübersetzt werden soll, findet der Precompiler über eine Fileselectorbox. Das dürfte angesichts der diversen Compiler der beste gemeinsame Weg sein. Wer möchte, kann dies natürlich ändern. In die Datei, aus der der Precompiler liest, schreibt er den erzeugten Text auch wieder zurück.

Offset	Register	
$00	RESPONSE	
$02	CONTROL	
$04	SAVE	
$06	RESTORE	
$08	OPERATION WORD	
$0A	COMMAND	
$0C	(RESERVED)	
$0E	CONDITION	
$10	OPERAND	
$14	REGISTER SELECT	(RESERVED)
$18	INSTRUCTION ADDRESS	

Tabelle 1: Coprozessor-Interface-Register

Auf diese Weise hat man es immer nur mit einem Modulatext zu tun und kommt nicht in Versuchung, in der vom Precompiler erzeugten Version schnell etwas zu ändern, was man dann nicht in die Stammversion übernimmt. Während der Arbeit schreibt der Precompiler in ein Meta-File, damit er jederzeit bei Fehlern abbrechen kann, ohne den Quelltext zu zerstören. Hat alles geklappt, wird der alte Quelltext gelöscht und das Meta-File umbenannt. Eine andere Lösung ist jedoch bei Bedarf auch leicht zu realisieren, ist doch das ganze Programm so ausgelegt, daß auch andere Modulaner leicht etwas anpassen können. Dafür wurde auf eine Totaloptimierung in Sachen Übersetzungsgeschwindigkeit verzichtet. Auf die Details werde ich bei der Besprechung des Programms eingehen.

Der harte Weg zur Hardware

Der Coprozessor steht dem Hauptprozessor des ATARI in seiner Komplexität in nichts nach. Die Dokumentation von Motorala um-

Verfügbare Operationen

Kürzel	Beschreibung
Fabs	produziert die positive Zahl
Facos	Arcus Cosinus
Fadd	Addition
Fasin	Arcus Sinus
Fatan	Arcus Tangens
Fatanh	Arcus Tangenshyperbolicus
Fcmp	Vergleiche
Fcos	Cosinus
Fcosh	Cosinushyperbolicus
Fdiv	Division Zieloperand durch Quelloperand nach Zieloperand
Fetox	e^x
Fetoxm1	$(e^x) - 1$
Fgetexp	Exponent der Zahl
Fgetman	Mantisse der Zahl
Fint	rundet auf die nächste ganze Zahl
Fintrz	rundet auf die nächste ganze Zahl ab
Flog10	Logarithmus zur Basis 10
Flog2	Logarithmus zur Basis 2
Flogn	natürlicher Logarithmus
Flognp1	Flogn(x+1)
Fmod	Modulofunktion, teilt und wirft das Ganzzahlige weg
Fmove	lädt eine Zahl oder Variable in die FPU oder den Hauptspeicher
Fmovecr	lädt eine Zahl aus dem Konstanten-ROM in ein Register
Fmul	Multiplikation
Fneg	Negation
Frem	Differenz des Zieloperanden zum nächstliegenden Vielfachen des Quelloperanden
Freset	Pseudooperation, setzt den Coprozessor in einen definierten Zustand
Fstatus	liest den Status in die INTEGER-Variable fpustatus
Fscale	addiert den zu einer Integerzahl gewandelten Quelloperanden zum binären Exponenten
Fsgldiv	Division mit halber Stellenzahl
Fsglmul	Multiplikation mit halber Stellenzahl
Fsin	Sinus
Fsincos	Sinus und Cosinus zugleich
Fsinh	Sinushyperbolicus
Fsqrt	Quadratwurzel
Fsub	Differenz
Ftan	Tangens
Ftanh	Tangenshyperbolicus
Ftentox	10^x
Ftst	Testen einer Zahl
Ftwotox	2^x

Tabelle 2: Aufstellung der implementierten Operationen

faßt mehrere hundert Seiten, doch zum Glück ist das meiste davon nur für Computerbauer und Betriebssystementwickler wichtig. Um den Precompiler einsetzen zu können, reicht es, die verfügbaren Datenformate und den Sprachschatz des Coprozessors zu kennen. Die Kommunikation zwischen 68000 und 68881 soll daher nur gestreift werden. Wer hier tiefer einsteigen möchte, sei auf [2] verwiesen bzw. möge mir schreiben. Im zweiten Teil wird dann darauf aufbauend erklärt, wie man mittels des Precompilers den Coprozessor in eigene Programme einbinden kann.

Die Architektur eines Chips

Im ATARI hat jeder Baustein eine Adresse, unter der ihn die anderen erreichen können. Der Coprozessor wird im Rechner über die Speicheradressen $FFFFFA40 bis $FFFFFA5B angesprochen, wo sich eine Reihe von Registern (Tabelle 1) befindet, über die die Kommunikation mit dem Hauptprozessor und den darauf laufenden Programmen stattfindet. Aufgrund dieser Funktion werden die Register auch Interface-Register genannt.

Für uns sind davon allerdings nur die interessant, bei denen die Anweisungen im Zusammenspiel mit einem 68020 sinnvoll einzusetzen sind.

Das Response-(Antwort-)Register gibt die Reaktion des Coprozessors auf die Anweisungen wieder, die er über das Command-(Anweisungs-)Register erhalten hat. Dort steht dann z.B., daß er noch Daten braucht, daß er noch nicht fertig ist, daß man das Kommunikationsprotokoll verletzt hat oder auch die Mitteilung, daß er zu weiteren Taten bereit ist.

Das Control-(Überwachungs-)Register hat beim 68881 nur die Funktion, daß ein Schreibzugriff auf den Coprozessor wie ein Reset wirkt. Er bricht dann alle Operationen ab und stellt klare Verhältnisse her.

Über das Condition-(Vergleichsmodus)-Register wird bei Vergleichsoperationen angegeben, in welcher Weise die Zahlen verglichen werden sollen.

Das neben dem Command-Register wohl wichtigste Register stellt schließlich das Operand-(Zahlenaustausch-)Register dar, über das die gesamten Fließkommazahlen vom Coprozessor zum 68000 bzw. in die Gegenrichtung strömen.

Bis hierher haben wir uns mit den "äußeren" Registern beschäftigt, die dem Hauptprozessor zugewandt sind. Jetzt soll ein Blick in die inneren Bereiche folgen, in denen die Fließkommazahlen, die über das Operand-Register hereingekommen

Kürzel	Gleichung	Bedeutung	
EQ	Z	Equal	Gleich
NE	$\overline{Z}$	Not Equal	Nicht Gleich
GT	$\overline{NAN \lor Z \lor N}$	Greater Than	Größer als
NGT	$NAN \lor Z \lor N$	Not Greater Than	Nicht Größer als
GE	$Z \lor \overline{(NAN \lor N)}$	Greater Than or Equal	Größer als oder gleich
NGE	$NAN \lor (N \land \overline{Z})$	Not (Greater Than or Equal)	Nicht (Größer als oder gleich)
LT	$N \land \overline{(NAN \lor Z)}$	Less Than	Kleiner als
NLT	$NAN \lor (Z \lor \overline{N})$	Not Less Than	Nicht weniger als
LE	$Z \lor (N \land \overline{NAN})$	Less Than or Equal	Kleiner als oder gleich
NLE	$NAN \lor \overline{(N \lor Z)}$	Not (Less Than or Equal)	Nicht (Kleiner als oder gleich)
GL	$\overline{NAN \lor Z}$	Greater or Less Than	Größer oder kleiner als
NGL	$NAN \lor Z$	Not(Greater or Less Than)	Nicht (Größer oder kleiner als)
GLE	$\overline{NAN}$	Greater, Less or Equal	Definiertes Verhältnis
NGLE	NAN	Not (Greater, Less or Equal)	Undefiniertes Verhältnis

Kürzel	Gleichung	Bedeutung	
OGT	$\overline{NAN \lor Z \lor N}$	Ordered Greater Than	Geordnet größer als
ULE	$NAN \lor Z \lor N$	Unordered or Less or Equal	Ungeordnet, kleiner oder gleich
OGE	$Z \lor \overline{(NAN \lor N)}$	Ordered Greater Than or Equal	Geordnet größer als oder gleich
ULT	$NAN \lor (N \land \overline{Z})$	Unordered or Less Than	Ungeordnet oder kleiner als
OLT	$N \land \overline{(NAN \lor Z)}$	Ordered Less Than	Geordnet kleiner als
UGE	$NAN \lor (Z \lor \overline{N})$	Unordered or Greater or Equal	Ungeordnet, größer oder gleich
OLE	$Z \lor (N \land \overline{NAN})$	Ordered Less Than or Equal	Geordnet kleiner als oder gleich
UGT	$NAN \lor \overline{(N \lor Z)}$	Unordered or Greater Than	Ungeordnet oder größer als
OGL	$\overline{NAN \lor Z}$	Ordered Greater or Less Than	Geordnet größer oder kleiner als
UEQ	$NAN \lor Z$	Unordered or Equal	Ungeordnet oder gleich
OR	$\overline{NAN}$	Ordered	Geordnet
UN	NAN	Unordered	Ungeordnet

Kürzel	Gleichung	Bedeutung	
F	*False*	False	Falsch
T	*True*	True	Richtig
SF	*False*	Signalling False	Falsch
ST	*True*	Signalling True	Richtig
SEQ	Z	Signalling Equal	Gleich
SNE	$\overline{Z}$	Signalling Not Equal	Nicht gleich

Tabelle 3: Vergleiche

sind, gespeichert und verarbeitet werden.

Da wären zunächst die acht Fließkommaregister *fp0* bis *fp7*, die jeweils eine Breite von 80 Bit haben. Die Fließkommazahlen werden dort in Form einer 64 Bit breiten Mantisse und einem 15 Bit langen Exponenten gespeichert. Ein weiteres Bit merkt sich das Vorzeichen der Mantisse. Eine 64 Bit breite Mantisse bedeutet, daß Fließkommazahlen mit einer Genauigkeit von 19 Stellen (!) gehandhabt werden und eine Mantisse von 15 Bit erlaubt die Darstellung von Zahlen im Bereich zwischen $8 * 10^{-4933}$ und $9 * 10^{4952}$. Dieses Format dürfte in aller Regel mehr als ausreichend sein.

Kürzel	Gleichung	Bedeutung
F	False	falsch
T	True	richtig
SF	False	falsch (signalling false)
ST	True	richtig (signalling true)
SEQ	Z	gleich (signalling equal)
SNE	$\overline{Z}$	nicht gleich (signalling not equal)

Tabelle 4

Da man im Speicher zumeist ein etwas platzsparenderes Format hat, werden Zahlen, die von außen kommen oder dorthin gelangen sollen, jeweils konvertiert. Der Nachfolger des 68881 - der 68882 - hat hierfür eine eigene Konvertereinheit (Conversion Execution Unit) und kann somit gleichzeitig rechnen und neue Zah-

MANche Leute HATTEN einmal einen ganz normalen ST......

.... jetzt gibt es die TETRA ST LINE, professionell und mega in.
Ein Beispiel ist der MANHATTEN ST –
Komponenten vom Feinsten
individuell assembliert

TOS 1.4, 2-4 MB, 3,5" und/oder 5,25" Floppy 720 K bis 1,44 MB, SCSI-Festplatten bis 380 MB, 28-15 msec, Wechselplatte 44 MB, gepufferte parallele Schnittstelle, NF-Verstärker mit eingebautem Lautsprecher, Zentralschalter, 220 W Netzteil und Flüsterlüfter, 5 m Tastaturspiralkabel, Software-Grundausstattung, Netzwerkkarte integrierbar, Monitore S/W, Color, Multisync 12"-21".
Bei der Assemblierung des MANHATTEN ST können vorhandene Komponenten integriert werden; Sonderwünsche und andere Gehäusetypen auf Anfrage.

Komponenten der TETRA ST LINE gibt es auch einzeln, z.B.:

Towergehäuse incl. 220 W Netzteil	DM 598,-
Komplettassemblierung o. Gehäuse	DM 580,-
16 MHz CPU mit Cache Memory	DM 598,-
PC-Emulator	DM 498,-
RAM-Erweiterung Mega ST2 auf 4 MB	DM 648,-
Floppy 5 1/4" 1,2 MB oder 3,5" 1,44 MB incl. Umschaltlogik und Software	DM 298,-
Festplatte 3,5" 50 MB 28 msec	DM 1398,-
Festplatte 3,5" 84 MB 24 msec	DM 1680,-
Festplatte 3,5" 170 MB 15 msec	DM 3330,-
14" S/W Multisync Monitor	DM 548,-
19" Großbildschirm	DM 3980,-
Netzwerke je Station	ab DM 1298,-

(alle Preise incl. Controller, Platinen, Einbau, zzgl. Versandkosten)

ab DM 4980,-
2 MB RAM, TOS 1.4
50 MB SCSI-Festplatte
14" S/W Monitor

TETRA
Computersysteme GmbH

Neuer Markt 27 5309 Meckenheim
☎ 0 22 25 - 1 70 81 Fax 0 22 25 - 1 70 83

T.S. Service

Szemere Hard & Software

Schleißheimerstr.127 8 Mü. 40

Tel. 089/3089408

Fax 089/3085636



Neu !!
Jetzt auch DDD Produkte
in München.......
Infos und Preisliste anfordern !!

ATARI

Festplattenumbau:
Megafile 30 auf 60 MB und leiser
DM 799.00

Sämtliche DDD Produkte jetzt auch in München erhältlich!!!

Info anfordern !!

Ab sofort auch PD's !!!

Take 20 / PD Pool

Original ATARI

1040 STF incl. Maus,Mon.und Basic DM 1259.00
1040 STE incl. Maus,Mon. und Basic DM 1559.00
Mega ST 2 incl. Maus,Mon. und Basic DM 2398.00
Megafile 30 DM 889.00
Megafile 60 DM 1479.00
Portfolio DM 598.00
NEU !!!
LYNX DM 379.00

Digital Data Deicke

Mega M 2 /PC incl. PC-SPEED Maus,Mon. und Basic DM 2759.00
Mega M 4/PC incl. PC-SPPED Maus,Mon. und Basic DM 3498.00
alle Rechner auch mit HYPERCACHE 16 Mhz a.A.
DDD 32 DM 899.00
DDD 64 DM 1354.00
DDD 62L neu DM 1389.00
Laser / 8 100% kompatibel mit Standby Funktion DM 2499.00

Zubehör

TOS 1.4 DM 189.00
PC-SPEED 1.40 DM 479.00
Speichererw. günstig a.A.
Speicherchips HYB TC 511000
1 MB DM 19.45
Hypercache DM 579.00
3,5" DM 249.00
5,25" Macrotec DM 269.00
Großbildschirm M110
Matrix DM 4199.00
Supercharger 1MB
Sonderaktion ! DM 699.00

Jetzt auch Software !!

Signum II	DM 399.00
1 st Word 3.15	DM 239.00
Adimens 3.0	DM 349.00
Calamus	DM 699.00
Lavadraw	DM 139.00

Laserdruckservice:

Wenn Sie mir Ihr Textfile zusenden,erhalten Sie in kürzester Zeit Ihre Ausdrucke per Post NN zurück.

Ideal für Doktor -und Magisterarbeiten.

in - Calamus -Signum II - 1 st Word -

Staffelpreise: Info anfordern

SpeedBridge & Laserinterface©

Es handelt sich hierbei um einen Steckadapter für den Einbau von PC-Speed Emulatoren in Rechner der Mega ST Serie. Die SpeedBridge wird auf den Mega-Systembus aufgesteckt, die Versorgungsleitung angeschlossen, die PC-Speed Platine wird nur noch eingesteckt und fertig. Es sind keinerlei Lötarbeiten erforderlich, so daß der Einbau auch für Ungeübte in weniger als 10 Minuten zu bewältigen ist. Fragen Sie Ihren Fachhändler nach der SpeedBridge oder wenden Sie sich direkt an uns, damit auch Sie zu den über 1000 zufriedenen SpeedBridge-Anwendern gehören.

gelb
orange
rot
braun
Pullup-Jumper
IPL1

unverbindlich empfohlene Verkaufspreise:

SpeedBridge	79.--
Laserinterface	89.--
HD-Kit extern	448.--
HD-Kit intern	398.--
Einbau HD-Kit kompl.	98.--

zzgl. 6.-- DM Porto- und Versandkosten

Unser Laserinterface ist ein Steckmodul für den Atari SLM 804 Laserdrucker, das den korrekten DMA-Betrieb mit abgeschalteten Laserdrucker ermöglicht und somit eine erhebliche Geräusch- und Stromverbrauchsreduzierung nach sich zieht. Das Steckmodul wird zwischen dem Laserdrucker und dem DMA-Treiber eingefügt und ist somit einfach nachrüstbar.

HD-Kit

Mit unserem HD-Kit ist es Ihnen möglich, mit 3,5 " HD-Disketten (1.44 Mb) zu arbeiten. Somit können Sie mit Disketten von IBM PS/2 - und Atari TT Systemen problemlos arbeiten. Das HD-Kit besteht aus einem anschlußfertigen TEAC FD 235 HF Laufwerk, einem Floppycontrollerboard und der entsprechenden Software zum Formatieren und Kopieren der HD-Disketten. Das HD-Kit ist für alle Rechner der Atari ST-Serie geeignet und macht Ihren Computer kompatibel für die Zukunft.

Vertrieb: digital image, Postfach 1206, D-6096 Raunheim, Tel.(06134) 51706 + 51748
in der Schweiz: DTZ Data Trade AG, Landstr. 1, CH-5415 Rieden/Baden, Tel.(056) 821880
in Österreich: Darius Computer, Andreas Huger Gasse 56/1, A-1220 Wien, Tel.(0222) 239580

len einlesen. Da so etwas allerdings nicht allzuoft benötigt wird, wenn man den Zahlentransfer von Hand regelt, sind die Vorteile des 68882 nicht besonders groß.

Die von der FPU unterstützten Datenformate:

B	8 Bit	BYTE
W	16 Bit	SHORTINT
L	32 Bit	LONGINT
S	32 Bit	REAL
D	64 Bit	LONGREAL
E	96 Bit	Internes Format mit weiteren 16 reservierten Bits
P	96 Bit	BCD-Format, für Textausgabe unter Fortran interessant. Hoher zeitlicher Aufwand bei der Umwandlung

Eine weitere interessante Struktur auf dem Chip ist das Konstanten-ROM. Dieser Festspeicher dient zum einen internen Zwecken und enthält Werte, die der Coprozessor für seine Arbeit benötigt. Da aber wohl noch etwas Platz war, sind auch etliche für den Programmierer nützliche Konstanten wie π oder e in das ROM aufgenommen worden.

Für die eigentliche Rechnerei hat die FPU dann weiterhin noch einen sogenannten Barrel Shifter, der in einem Taktzyklus beliebige Verschiebungen durchführen kann, interne temporäre Register, einen weiteren Shifter, eine Rundungslogik und schließlich die alles überragende Arithmetiklogikeinheit (ALU), die die Zahlen dann letztendlich "verwürfelt". Diese Teile sind dem Benutzer jedoch verschlossen und sollten hier nur der Vollständigkeit halber erwähnt werden.

Die Sprache der Mathematik

Nachdem die Architektur des 68881 beschrieben wurde, wenden wir uns jetzt der Darstellung dessen zu, was man aus diesem Baustein so "herauskitzeln" kann - und das ist eine ganze Menge.

Da gibt es zunächst einige Anweisungen, die im Zusammenspiel mit einem 68020 dessen vorhandene Möglichkeiten auf Fließkommazahlen hin erweitern:

FBcc:	Branch Conditionally, bedingter Sprung
FDBcc:	Test Condition, Decrement and Branch, Dekrementieren und bedingt springen
FTRAPcc:	Trap Conditionally, bedingte Ausnahmeverarbeitung
FScc:	Set According to Condition, umstandsgemäß setzen
FNOP:	No Operation, keine Operation

Daneben gibt es Operationen, die speziell für Multitasking-Umgebungen gedacht sind:

FMOVEM:	Kontextwechsel der Fließkommaregister
FSAVE:	sichert den momentanen Status der gesamten FPU mit allen "geheimen" Daten
FRESTORE:	stellt den alten Zustand wieder her

Dem eingefleischten Assembler-Programmierer mag hier das Herz aufgehen. Die Kombination 68020 und 68881 stellt sich ihm wie ein einziger Baustein dar, die Mauer zwischen den natürlichen und den rationalen Zahlen ist gebrochen. Da diese beiden Gruppen für den anwendungsorientierten Programmierer jedoch wenig sinnvoll sind, habe ich mir auch nicht die Arbeit gemacht, sie in meinen Precompiler mit aufzunehmen.

Das Salz in der Suppe sind die eigentlichen Arithmetikoperationen, die den größten Teil darstellen und vollständig vom Precompiler unterstützt werden. Hier findet sich wirklich alles, was man irgendwie gebrauchen kann, angefangen bei diversen Logarithmen bis hin zu Area tangens hyperbolicus.

Jede Operation findet zwischen zwei Operanden statt, dem Quell- und dem Zieloperand. Diese können auch identisch sein, wenn man beispielsweise von der Zahl in Register eins den Sinus berechnen und in dasselbe legen läßt. Wissen muß man lediglich, daß der Quelloperand der Subtrahend bzw. der Divisor ist, da bei diesen beiden Operationen die Reihenfolge eine Rolle spielt. Die Rechenoperationen finden entweder zwischen zwei Registern statt oder der Quelloperand stammt aus dem Speicher. Soll ein Wert wieder in den Speicher kommen, ist hierfür eine explizite MOVE-Anweisung nötig. Eine vollständige Auflistung der mathematischen Anweisungen ist in Tabelle 2 zu sehen.

Hier sollen nur diejenigen vorgestellt werden, die etwas außergewöhnlich sind und einer näheren Beschreibung bedürfen.

FMOVECR:

lädt eine Konstante aus dem Konstanten-ROM in eines der Fließkommaregister. Insgesamt hat das ROM 64 Einträge, von denen jedoch nur 22 für den Programmierer gedacht und damit dokumentiert sind. Bei den anderen können von FPU zu FPU Unterschiede auftreten, je nach Version des Coprozessors. Von einer Verwendung der restlichen 42 Konstanten muß also im Interesse der Übertragbarkeit der Programme abgeraten werden, sofern sich dabei überhaupt interessante Werte finden lassen. Eine vollständige Liste der 22 freigegebenen Konstanten findet sich in Tabelle 5.

REM:

Diese Funktion gibt die Differenz des Zieloperanden zum nächstliegenden Vielfachen des Quelloperanden an. Die Zahl kann auch ein negatives Vorzeichen haben, je nach Lage der Dinge:

REM(0.9, 0.5) = -0.4
REM(0.5, 0.9) = -0.1
REM(0.5, 1.6) = 0.1

SGLDIV/SGLMUL:

Division bzw. Multiplikation mit der Genauigkeit einer REAL-Variablen. Der Geschwindigkeitsgewinn beträgt bei der Division 2,5 µs, falls sie zwischen Registern stattfindet, sonst 2,0 µs. Bei der Multiplikation zweier Register ist SGLMUL eine halbe µs schneller. Ist der Speicher beteiligt, hat man keinen Geschwindigkeitsvorteil.

SCALE:

wandelt den Quelloperanden in eine Integerzahl (falls nötig) und zählt ihn dann zum binären Exponenten des Zieloperanden. Diese Operation verhält sich von der Geschwindigkeit her wie SGLMUL.

SINCOS:

Diese Funktion berechnet gleichzeitig den Sinus und den Cosinus eines Werts. Der zusätzliche Zeitbedarf beträgt 15% der Rechenzeit einer Einzelberechnung. Für das Rechnen mit komplexen Zahlen recht nützlich.

CMP:

compare, vergleiche. Diese Funktion führt einen Vergleich zweier Zahlen durch, wobei man ihr noch mitteilt, was man letztlich feststellen möchte, z.B. ob die zweite Zahl größer ist. Aus dem oben erwähnten Response-Register erhält man dann

Konstanten

Nummer	Konstante
$00	π
$11	$\log_{10}(2)$
$12	e
$13	$\log_2(e)$
$14	$\log_{10}(e)$
$15	0.0
$48	ln(2)
$49	ln(10)
$50	10^0
$51	10^1
$52	10^2
$53	10^4
$54	10^8
$55	10^{16}
$56	10^{32}
$57	10^{64}
$58	10^{128}
$59	10^{256}
$60	10^{512}
$61	10^{1024}
$62	10^{2048}
$63	10^{4096}

Tabelle 5: Aufstellung der für den Benutzer gedachten Anweisungen

die Antwort "ja" oder "nein". Die Reihe der möglichen Vergleiche ist ausgesprochen lang, wobei aber ein guter Teil äquivalent ist, sofern man keinen 68020 hat und z.B. Überläufe mit Traps quittieren läßt. Die Auflistung befindet sich in Tabelle 3. Die drei Tabellen enthalten eine Aufstellung der verfügbaren Vergleichsmöglichkeiten. Die in den Boolschen Gleichungen verwendeten Kürzel *NAN*, *N* und *Z* bedeuten "Keine Zahl" (Not a Number), "Negativ" und "Null" (Zero). Daraus leiten sich alle Vergleichsbedingungen ab. Wie man an den logischen Gleichungen sieht, sind die Vergleichsbedingungen der zweiten Tabelle bereits unter einem anderen Namen in der ersten Tabelle realisiert. Die Unterschiede sind für uns ohne Belang. Am besten wählt man die Variante aus, die am einprägsamsten ist. Die Einträge der dritten Tabelle wird man lediglich als Dummy zum Setzen oder Löschen der Boolschen Variable verwenden können. Motorola schreibt selbst, daß sie eigentlich nur der Vollständigkeit halber implementiert wurden.

TST:

Diese Operation ist der zweite Teil von CMP. CMP vergleicht zwei Zahlen, in dem es zunächst die erste von der zweiten subtrahiert und dann anhand des später verworfenen Ergebnisses seine Schlüsse zieht. Dies ist natürlich auch möglich, ohne daß man den Subtrahenden und den Minuenden kennt und genau das macht TST.

Soviel für diesmal, nächsten Monat wird es dann um den Aufbau und die Benutzung des Precompilers gehen, wofür dieser Artikel die Grundlagen gelegt hat. Wer will kann ja schon einmal den ersten Teil des Listings eintippen, das dann nächsten Monat fortgesetzt wird.

Jens Decker

Literatur:

[1] *M. Kofler: "Apfelmännchen mit 68881-Speed", ST Computer 4/89, S. 150ff*

[2] *Motorola: MC68881/MC68882 Floating-Point Coprozessor User's Manual, Prentice Hall, 1987*

```
 1: MODULE FPUPreprozessor ;   (* Für Hänisch Modula
 2:                                  V3.105 *)
 3: (* Projekt        : FPUPreprozessor *)
 4: (* Teil           : Hauptmodul *)
 5: (* Weitere Files  : FPU_PREP.TXT Vorarbeiten für
                        den FPUPreprozessor *)
 6: (*                  FPU_MAKE.INF MakeFile für
                        Hänisch Modula *)
 7: (* Beginn         : 07.07.89 *)
 8: (* akt. Datum     : 02.05.90 *)
 9: (* akt. Arbeiten  : Anpassung an das Megamax
                        Modula *)
10: (* by Jens Decker *)
11: (* Copyright: MAXON Computer GmbH *)
12:
13: FROM AESApplications      IMPORT ApplInit,
                                     ApplExit ;
14: FROM AESForms             IMPORT FormAlert ;
15: FROM BufInOut             IMPORT FilterBase,
                NumFilter, RDone, ReadLine, WriteLn,
16:             WriteString ;
17: FROM Conversions          IMPORT StrToVal,
                                     ValToStr ;
18: FROM DOSbase              IMPORT tFilename,
                                     tPathname ;
19: FROM Directories          IMPORT Delete, Rename ;
20: FROM FSelOpen             IMPORT FSelOpen,
                                     DummyOpenErr ;
21: FROM Paths                IMPORT GetCurrentPath,
                              GetPurePath, GetFile ;
22: FROM GEMAESBase           IMPORT tExitButton,
                                     tFSelReturn ;
23: FROM Streams              IMPORT Close, Open,
                         tStreamHandle, tStreamMode ;
24: FROM String               IMPORT Concat, Copy,
                                     Length, Pos ;
25: FROM SYSTEM               IMPORT ADR, ADDRESS ;
26: FROM RealConversions      IMPORT StrToReal ;
27: IMPORT InOut ;
28:
29: TYPE Compilertyp = (HaenischM, PDM, TDI, SPC,
                        Megamax, noComp) ;
30:      Operationstyp = (move, movecr, int, sinh,
                         intrz, sqrt, lognp1, etoxm1,
31:                      tanh, atan, asin, atanh,
                         sin, tan, etox, twotox,
32:                      tentox, logn, log10, log2,
                         abs, cosh, neg, acos,
33:                      cos, getexp, getman, div,
                         mod, add, mul, sgldiv,
34:                      rem, reset, status, scale,
                         sglmul, sub, sincos, cmp,
35:                      tst, noop) ;
36:      Extensiontyp  = (L, S, E, P, W, D, B, Space,
                          Klammer, noext) ;
37:                     (* P wird nicht unterstützt *)
38:      Uebergabetyp  = (Num, Var, Reg0, Reg1, Reg2,
                            Reg3, Reg4, Reg5,
39:                         Reg6, Reg7, nouebg) ;
                    (* Eventuell noch Konstanten! *)
40:      Vergleichstyp = (F, EQ, OGT, OGE, OLT, OLE,
                         OGL, ORd, UN, UEQ, UGT,
41:                      UGE, ULT, ULE, NE, T, SF,
                         SEQ, GT, GE, LT,LE, GL,
42:                      GLE, NGLE, NGL, NLE, NLT,
                         NGE, NGT, SNE, ST,novgl) ;
43:      tfpuop = RECORD
44:                 Operation     : Operationstyp ;
45:                 Extension     : Extensiontyp ;
46:                 Argument1     : Uebergabetyp ;
47:                 Argument2     : Uebergabetyp ;
48:                 Argument3     : Uebergabetyp ;
49:                 Argument4     : Vergleichstyp ;
50:                 VarNum        : ARRAY [0..79] OF
                                    CHAR ;
51:                 Boolean       : ARRAY [0..79] OF
                                    CHAR ;
52:                 Zeile         : ARRAY [0..511] OF
                                    CHAR ;
53:                 Zeilennummer  : INTEGER ;
54:                 Zeiger        : SHORTCARD ;
55:                 ZeigerAnfang  : SHORTCARD ;
56:               END ;
57:
58: VAR ApID      : SHORTINT ;
59:     fpuop     : tfpuop ;
60:     Compiler  : Compilertyp ;
61:
62: PROCEDURE Intro ; (* Kann für die tägliche
                  Arbeit auskommentiert werden *)
63: VAR AlertReturn : tExitButton ;
64:     AlertText   : ARRAY [1..120] OF CHAR ;
65: BEGIN
66:   AlertText :=
67:   "[0][FPU-Precompiler V1.1 |1989 by Jens
       Decker |Fichtenweg 7/108 |7400
       Tübingen ][ OK ]" ;
68:    AlertReturn := FormAlert(FirstExit,
                                AlertText) ;
69: END Intro ;
70:
71: PROCEDURE FehlerMeldung(Text : ARRAY OF CHAR) ;
72: VAR AlertReturn : tExitButton ;
73:     AlertText   : ARRAY [0..150] OF CHAR ;
74:     ValStr      : ARRAY [0..5] OF CHAR ;
75:     Error       : BOOLEAN ;
76: BEGIN
77:   AlertText := "[1][ Zeile " ;
78:   ValToStr(fpuop.Zeilennummer, FALSE, ValStr, 10,
               5, "_", Error) ;
79:   Concat(AlertText, ValStr) ;
80:   Concat(AlertText, ": |") ;
81:   Concat(AlertText, Text) ;
82:   Concat(AlertText, " ][ Abbruch ]") ;
83:   AlertReturn := FormAlert(FirstExit,AlertText);
84: END FehlerMeldung ;
85:
86: MODULE StreamHandling ;
87:   IMPORT FSelOpen, DummyOpenErr, GetCurrentPath,
88:          GetPurePath, GetFile, Close, Open,
             Delete, Rename, Concat,
89:          tFilename, tFSelReturn, tPathname,   →
```

```
 90:          tStreamHandle, tStreamMode ;
 91:     EXPORT OeffneStreams, SchliesseStreams ;
 92:
 93:     VAR FileName     : tFilename ;
 94:         PathName     : tPathname ;
 95:         FSelReturn   : tFSelReturn ;
 96:         InputFile,
 97:         OutputFile   : tStreamHandle ;
 98:         ok           : BOOLEAN ; (* Dummy *)
 99:
100:     PROCEDURE OeffneStreams() : BOOLEAN ;
101:     BEGIN
102:       InputFile := FSelOpen(PathName, "*.MOD",
                       "Modula-Text auswählen!",
103:                    DummyOpenErr, ReadOnly, 512) ;
104:       IF InputFile = -1
105:         THEN RETURN FALSE ;
106:       END ;
107:       GetFile(PathName, FileName) ;
108:       GetPurePath(PathName, PathName) ;
109:       Concat(PathName, "METAFILE") ;
110:       OutputFile := Open(PathName,WriteOnly, 512);
111:       IF OutputFile = -1
112:         THEN RETURN FALSE ;
113:       END ;
114:       RETURN TRUE ;
115:     END OeffneStreams ;
116:
117:     PROCEDURE SchliesseStreams ;
118:     VAR PathMetaFile : tPathname ;
119:     BEGIN
120:       Close(InputFile) ;
121:       Close(OutputFile) ;
122:       GetPurePath(PathName, PathName) ;
123:       Concat(PathName, FileName) ;
124:       ok := Delete(PathName) ;
125:       GetPurePath(PathName, PathMetaFile) ;
126:       Concat(PathMetaFile, "METAFILE") ;
127:       ok := Rename(PathMetaFile, PathName) ;
128:     END SchliesseStreams ;
129:
130:   END StreamHandling ;
131:
132:   MODULE Parser ;
133:     IMPORT fpuop, tfpuop, Compilertyp,
              Operationstyp, Extensiontyp,
134:          Uebergabetyp, Vergleichstyp, Compiler,
135:          RDone, FehlerMeldung, ReadLine, WriteLn,
              WriteString, NumFilter,
136:          FilterBase, Pos, Length, Copy, InOut ;
137:     EXPORT ParserInit, Compileridentifizieren,
              Fsuchen, SucheEnde,
138:          Operationidentifizieren, CompIDs,
139:          Extensionidentifizieren, SucheOperand1,
              OperandenIdentifizieren,
140:          Syntaxpruefung, Operationen, Extensions,
              Operanden, Vergleiche, Klammerauf,
141:          Klammerzu ;
142:
143:     TYPE SMALLSTRING = ARRAY [0..7] OF CHAR ;
144:
145:     VAR Operationen : ARRAY [ORD(move)..ORD(noop)]
                           OF SMALLSTRING ;
146:         Extensions  : ARRAY [ORD(L)..ORD(noext)]
                           OF SMALLSTRING ;
147:         Operanden   : ARRAY
                           [ORD(Num)..ORD(nouebg)] OF
                           SMALLSTRING ;
148:         Vergleiche  : ARRAY [ORD(F)..ORD(novgl)]
                           OF SMALLSTRING ;
149:         CompIDs     : ARRAY [HaenischM..noComp] OF
150:                       ARRAY [1..20] OF CHAR ;
151:         Klammerauf,
152:         Klammerzu   : ARRAY [0..1] OF CHAR ;
153:         Klauf, Klzu : ARRAY [0..1] OF CHAR ;
154:         Komma       : ARRAY [0..1] OF CHAR ;
155:         Semikolon   : ARRAY [0..1] OF CHAR ;
156:
157:     PROCEDURE ParserInit ;
158:     BEGIN
159:       FilterBase := 10 ;
160:       Klammerauf := "(*" ;
161:       Klammerzu := "*)" ;
162:       Klauf := "(" ;
163:       Klzu := ")" ;
164:       Komma := "," ;
165:       Semikolon := ";" ;
166:       fpuop.Zeilennummer := 0 ;
167:       Operationen[ORD(move)] := "move" ;
168:       Operationen[ORD(movecr)] := "movecr" ;
169:       Operationen[ORD(int)] := "int" ;
170:       Operationen[ORD(sinh)] := "sinh" ;
171:       Operationen[ORD(intrz)] := "intrz" ;
172:       Operationen[ORD(sqrt)] := "sqrt" ;
173:       Operationen[ORD(lognp1)] := "lognp1" ;
174:       Operationen[ORD(etoxm1)] := "etoxm1" ;
175:       Operationen[ORD(tanh)] := "tanh" ;
176:       Operationen[ORD(atan)] := "atan" ;
177:       Operationen[ORD(asin)] := "asin" ;
178:       Operationen[ORD(atanh)] := "atanh" ;
179:       Operationen[ORD(sin)] := "sin" ;
180:       Operationen[ORD(tan)] := "tan" ;
181:       Operationen[ORD(etox)] := "etox" ;
182:       Operationen[ORD(twotox)] := "twotox" ;
183:       Operationen[ORD(tentox)] := "tentox" ;
184:       Operationen[ORD(logn)] := "logn" ;
185:       Operationen[ORD(log10)] := "log10" ;
186:       Operationen[ORD(log2)] := "log2" ;
187:       Operationen[ORD(abs)] := "abs" ;
188:       Operationen[ORD(cosh)] := "cosh" ;
189:       Operationen[ORD(neg)] := "neg" ;
190:       Operationen[ORD(acos)] := "acos" ;
191:       Operationen[ORD(cos)] := "cos" ;
192:       Operationen[ORD(getexp)] := "getexp" ;
193:       Operationen[ORD(getman)] := "getman" ;
194:       Operationen[ORD(div)] := "div" ;
195:       Operationen[ORD(mod)] := "mod" ;
196:       Operationen[ORD(add)] := "add" ;
197:       Operationen[ORD(mul)] := "mul" ;
198:       Operationen[ORD(sgldiv)] := "sgldiv" ;
199:       Operationen[ORD(rem)] := "rem" ;
200:       Operationen[ORD(reset)] := "reset" ;
201:       Operationen[ORD(status)] := "status" ;
202:       Operationen[ORD(scale)] := "scale" ;
203:       Operationen[ORD(sglmul)] := "sglmul" ;
204:       Operationen[ORD(sub)] := "sub" ;
205:       Operationen[ORD(sincos)] := "sincos" ;
206:       Operationen[ORD(cmp)] := "cmp" ;
207:       Operationen[ORD(tst)] := "tst" ;
208:       Operationen[ORD(noop)] := "" ;
209:       Extensions[ORD(L)] := "L" ;
210:       Extensions[ORD(S)] := "S" ;
211:       Extensions[ORD(E)] := "E" ;
212:       Extensions[ORD(P)] := "P" ;
213:       Extensions[ORD(W)] := "W" ;
214:       Extensions[ORD(D)] := "D" ;
215:       Extensions[ORD(B)] := "B" ;
216:       Extensions[ORD(Space)] := " " ;
217:       Extensions[ORD(Klammer)] := "(" ;
218:       Extensions[ORD(noext)] := "" ;
219:       Operanden[ORD(Num)] := "Num" ;
220:       Operanden[ORD(Var)] := "Var" ;
221:       Operanden[ORD(Reg0)] := "fp0" ;
222:       Operanden[ORD(Reg1)] := "fp1" ;
223:       Operanden[ORD(Reg2)] := "fp2" ;
224:       Operanden[ORD(Reg3)] := "fp3" ;
225:       Operanden[ORD(Reg4)] := "fp4" ;
226:       Operanden[ORD(Reg5)] := "fp5" ;
227:       Operanden[ORD(Reg6)] := "fp6" ;
228:       Operanden[ORD(Reg7)] := "fp7" ;
229:       Operanden[ORD(nouebg)] := "" ;
230:       Vergleiche[ORD(F)] := "F" ;
231:       Vergleiche[ORD(EQ)] := "EQ" ;
232:       Vergleiche[ORD(OGT)] := "OGT" ;
233:       Vergleiche[ORD(OGE)] := "OGE" ;
234:       Vergleiche[ORD(OLT)] := "OLT" ;
235:       Vergleiche[ORD(OLE)] := "OLE" ;
236:       Vergleiche[ORD(OGL)] := "OGL" ;
237:       Vergleiche[ORD(ORd)] := "OR" ;
238:       Vergleiche[ORD(UN)] := "UN" ;
239:       Vergleiche[ORD(UEQ)] := "UEQ" ;
240:       Vergleiche[ORD(UGT)] := "UGT" ;
241:       Vergleiche[ORD(UGE)] := "UGE" ;
242:       Vergleiche[ORD(ULT)] := "ULT" ;
243:       Vergleiche[ORD(ULE)] := "ULE" ;
244:       Vergleiche[ORD(NE)] := "NE" ;
245:       Vergleiche[ORD(T)] := "T" ;
246:       Vergleiche[ORD(SF)] := "SF" ;
247:       Vergleiche[ORD(SEQ)] := "SEQ" ;
248:       Vergleiche[ORD(GT)] := "GT" ;
249:       Vergleiche[ORD(GE)] := "GE" ;
250:       Vergleiche[ORD(LT)] := "LT" ;
```

→

```
251:      Vergleiche[ORD(LE)] := "LE" ;
252:      Vergleiche[ORD(GL)] := "GL" ;
253:      Vergleiche[ORD(GLE)] := "GLE" ;
254:      Vergleiche[ORD(NGLE)] := "NGLE" ;
255:      Vergleiche[ORD(NGL)] := "NGL" ;
256:      Vergleiche[ORD(NLE)] := "NLE" ;
257:      Vergleiche[ORD(NLT)] := "NLT" ;
258:      Vergleiche[ORD(NGE)] := "NGE" ;
259:      Vergleiche[ORD(NGT)] := "NGT" ;
260:      Vergleiche[ORD(SNE)] := "SNE" ;
261:      Vergleiche[ORD(ST)] := "ST" ;
262:      Vergleiche[ORD(novgl)] := "" ;
263:      CompIDs[HaenischM] := "Hänisch" ;
264:      CompIDs[PDM] := "TU München" ;
265:      CompIDs[TDI] := "TDI" ;
266:      CompIDs[SPC] := "SPC" ;
267:      CompIDs[Megamax] := "Megamax" ;
268:      CompIDs[noComp] := "" ;
269:    END ParserInit ;
270:
271:    PROCEDURE Compileridentifizieren() : BOOLEAN ;
272:    VAR posComp  : SHORTINT ;
273:    BEGIN
274:      REPEAT
275:        Compiler := HaenischM ;
276:        REPEAT
277:          posComp := Pos(fpuop.Zeile,
                 CompIDs[Compiler], fpuop.Zeiger);
278:          INC(Compiler) ;
279:        UNTIL (posComp # -1) OR (Compiler =
                                   noComp) ;
280:        DEC(Compiler) ;
281:
282:        IF posComp = -1
283:          THEN WriteString(fpuop.Zeile) ; WriteLn ;
284:               ReadLine(fpuop.Zeile) ;
285:               INC(fpuop.Zeilennummer) ;
286:               fpuop.Zeiger := 0 ;
287:          ELSE fpuop.Zeiger := posComp +
                     Length(CompIDs[Compiler]) ;
288:        END ;
289:      UNTIL (RDone = FALSE) OR (posComp # -1) ;
290:      IF posComp = -1
291:        THEN FehlerMeldung("Kein Compiler |
                           definiert!") ;
292:             RETURN FALSE ;
293:        ELSE RETURN TRUE ;
294:      END ;
295:    END Compileridentifizieren ;
296:
297:    PROCEDURE Fsuchen() : BOOLEAN ;
298:    VAR positionF,
299:        positionKlammerauf : SHORTINT ;
300:        resultat           : BOOLEAN ;
301:        F                  : CHAR ;
302:    BEGIN
303:      F := "F" ;
304:      positionF := Pos(fpuop.Zeile, F,
                         fpuop.Zeiger) ;
305:      positionKlammerauf := Pos(fpuop.Zeile,
                         Klammerauf, fpuop.Zeiger) ;
306:
307:      IF ((positionF < positionKlammerauf) AND
             (positionF # -1)) OR
308:         ((positionF # -1) AND (positionKlammerauf
                                          = -1))
309:        THEN fpuop.Zeiger :=positionF + Length(F);
310:             resultat := TRUE ;
311:      ELSIF ((positionKlammerauf < positionF) AND
              (positionKlammerauf # -1)) OR
312:            ((positionKlammerauf # -1) AND
              (positionF = -1))
313:        THEN fpuop.Zeiger := positionKlammerauf +
                               Length(Klammerauf) ;
314:             resultat := Klammernumgehen() ;
                           (* FALSE = F E H L E R ! *)
315:             IF resultat
316:               THEN resultat := Fsuchen() ;
317:               ELSE FehlerMeldung(" Ein Kommentar
                    wurde | nicht geschlossen!") ;
318:             END ;
319:      ELSIF ((positionF = -1) AND
             (positionKlammerauf = -1))
320:        THEN WriteString(fpuop.Zeile) ; WriteLn ;
321:             ReadLine(fpuop.Zeile) ;
322:             INC(fpuop.Zeilennummer) ;
323:             fpuop.Zeiger := 0 ;
324:             IF RDone
325:               THEN resultat := Fsuchen() ;
326:               ELSE resultat := FALSE ;
327:             END ;
328:      END ;
329:      IF resultat
330:        THEN fpuop.ZeigerAnfang :=
                               fpuop.Zeiger - 1 ;
331:      END ;
332:      RETURN resultat ;
333:    END Fsuchen ;
334:
335:    PROCEDURE Klammernumgehen() : BOOLEAN ;
336:    VAR positionKlammerzu,
337:        positionKlammerauf : SHORTINT ;
338:        Klammerebene       : SHORTCARD ;
339:    BEGIN
340:      Klammerebene := 1 ;
341:      REPEAT
342:        positionKlammerzu := Pos(fpuop.Zeile,
                   Klammerzu, fpuop.Zeiger) ;
343:        positionKlammerauf := Pos(fpuop.Zeile,
                   Klammerauf, fpuop.Zeiger) ;
344:
345:        IF ((positionKlammerzu <
                positionKlammerauf) AND
                (positionKlammerzu # -1)) OR
346:              ((positionKlammerzu # -1) AND
                (positionKlammerauf = -1))
347:          THEN fpuop.Zeiger := positionKlammerzu +
                               Length(Klammerzu) ;
348:               Klammerebene := Klammerebene - 1 ;
349:        ELSIF ((positionKlammerauf <
                 positionKlammerzu) AND
                 (positionKlammerauf # -1)) OR
350:              ((positionKlammerauf # -1) AND
                (positionKlammerzu = -1))
351:          THEN fpuop.Zeiger := positionKlammerauf +
                               Length(Klammerauf) ;
352:               Klammerebene := Klammerebene + 1 ;
353:        ELSIF ((positionKlammerauf = -1) AND
                 (positionKlammerzu = -1))
354:          THEN WriteString(fpuop.Zeile) ; WriteLn ;
355:               ReadLine(fpuop.Zeile) ;
356:               INC(fpuop.Zeilennummer) ;
357:               fpuop.Zeiger := 0 ;
358:               IF NOT RDone
359:                 THEN RETURN FALSE ;
                    (* Fehlermeldung in Fsuchen() *)
360:               END ;
361:        END ;
362:      UNTIL Klammerebene = 0 ;
363:    RETURN TRUE ;
364:    END Klammernumgehen ;
365:
366:    PROCEDURE SucheEnde ;
367:    VAR Position1 : SHORTINT ;
368:    BEGIN
369:      Semikolon := ";" ;
370:      Position1 := Pos(fpuop.Zeile, Semikolon,
                   fpuop.Zeiger) ;
371:      IF Position1 # -1
372:        THEN fpuop.Zeiger := Position1 + 1 ;
373:      END ;
374:    END SucheEnde ;
375:
376:    PROCEDURE Operationidentifizieren() : BOOLEAN ;
377:    VAR Operation    : Operationstyp ;
378:        OperationsNr : SHORTCARD ;
379:        Laenge       : SHORTCARD ;
380:    BEGIN
381:      Operation := move ;
382:      OperationsNr :=
              IdentifiziereString(fpuop.Zeile,
                  fpuop.Zeiger, Operationen,
383:      ORD(move), ORD(noop)) ;
384:      INC(Operation, OperationsNr) ;
385:      IF Operation = noop
386:        THEN RETURN FALSE ;
387:        ELSE fpuop.Operation := Operation ;
388:             Laenge :=
             Length(Operationen[ORD(Operation)]) ;
389:             fpuop.Zeiger:= fpuop.Zeiger + Laenge;
390:             RETURN TRUE ;
391:      END ;
392:    END Operationidentifizieren ;
393:
```

→

```
394:    PROCEDURE Extensionidentifizieren() : BOOLEAN ;
395:    VAR Extension   : Extensiontyp ;
396:        ExtensionNr : SHORTCARD ;
397:        Laenge      : SHORTCARD ;
398:    BEGIN
399:      Extension := L ;
400:      ExtensionNr :=
                IdentifiziereString(fpuop.Zeile,
                       fpuop.Zeiger, Extensions,
401:      ORD(Extension), ORD(noext)) ;
402:      INC(Extension, ExtensionNr) ;
403:      IF (Extension = noext) OR (Extension = P)
          (* P nicht implementiert *)
404:        THEN FehlerMeldung(" Diese Extension ist |
                           nicht implementiert") ;
405:             RETURN FALSE ;
406:      ELSIF (Extension = Space) OR (Extension =
                                      Klammer)
407:        THEN fpuop.Extension := D ;
408:        ELSE fpuop.Extension := Extension ;
409:      END ;
410:      Laenge:= Length(Extensions[ORD(Extension)]);
411:      IF Extension = Klammer
412:        THEN DEC(Laenge) ;
413:      END ;
414:      fpuop.Zeiger := fpuop.Zeiger + Laenge ;
415:      RETURN TRUE ;
416:    END Extensionidentifizieren ;
417:
418:    PROCEDURE SucheOperand1() : BOOLEAN ;
419:    VAR Anfang, Ende : SHORTINT ;
420:    BEGIN
421:      Anfang := Pos(fpuop.Zeile, Klauf,
                   fpuop.Zeiger) ;
422:      Ende := Pos(fpuop.Zeile, Klzu,fpuop.Zeiger);
423:      IF Anfang = Pos(fpuop.Zeile, Klammerauf,
                     fpuop.Zeiger)
424:        THEN Anfang := -1 ;
425:      END ;
426:      IF Ende = Pos(fpuop.Zeile, Klammerzu,
                   fpuop.Zeiger)
427:        THEN Ende := -1 ;
428:      END ;
429:      IF (Anfang = -1) OR (Ende = -1)
430:        THEN FehlerMeldung(" Operandenliste
                             fehlt!") ;
431:             RETURN FALSE ;
432:      ELSIF Anfang > Ende
433:        THEN FehlerMeldung(" '(' fehlt!") ;
434:             RETURN FALSE ;
435:        ELSE fpuop.Zeiger := Anfang + 1 ;
436:             WHILE fpuop.Zeile[fpuop.Zeiger] <=
                 " " DO
437:               INC(fpuop.Zeiger) ;
438:             END ;
439:             IF fpuop.Zeile[fpuop.Zeiger] = ")"
440:               THEN FehlerMeldung(" Operandenliste
                                       fehlt!") ;
441:                    RETURN FALSE ;
442:             END ;
443:      END ;
444:      RETURN TRUE ;
445:    END SucheOperand1 ;
446:
447:    PROCEDURE OperandenIdentifizieren() : BOOLEAN ;
448:    VAR Operand     : Uebergabetyp ;
449:        OperandNr   : SHORTCARD ;
450:        VergleichNr : SHORTCARD ;
451:        Laenge      : SHORTCARD ;
452:        Position1,
453:        Position2   : SHORTINT ;
454:        Freiraum    : ARRAY [0..1] OF CHAR ;
455:        Weitermachen: BOOLEAN ;
456:
457:      PROCEDURE TestRegVar(VAR Argumentx :
                             Uebergabetyp ;
458:                       VAR Ablage : ARRAY OF
                             CHAR) : BOOLEAN ;
459:      BEGIN
460:        Argumentx := Reg0 ;
461:        OperandNr :=
                IdentifiziereString(fpuop.Zeile,
                        fpuop.Zeiger, Operanden,
462:        ORD(Argumentx), ORD(nouebg)) ;
463:        INC(Argumentx,OperandNr - ORD(Argumentx));
464:        IF Argumentx < nouebg
465:          THEN Laenge :=
                      Length(Operanden[ORD(Argumentx)]) ;
466:                      fpuop.Zeiger := fpuop.Zeiger +
                              Laenge ;
467:        ELSIF ChIsLetter(fpuop.Zeile[fpuop.Zeiger])
468:          THEN Freiraum := " " ;
469:  (*               Position1 := Pos(fpuop.Zeile,
                        Freiraum, fpuop.Zeiger) ; *)
470:                   Position1 := Pos(fpuop.Zeile,
                              Komma, fpuop.Zeiger) ;
471:  (*               IF ((Position2 < Position1) AND
                 (Position2 # -1)) OR (Position1 = -1)
472:                   THEN Position1 := Position2 ;
473:                 END ; *)
474:                 Position2 := Pos(fpuop.Zeile,
                              Klzu, fpuop.Zeiger) ;
475:                 IF ((Position2 < Position1) AND
                     (Position2 # -1)) OR
                     (Position1 = -1)
476:                   THEN Position1 := Position2 ;
477:                 END ;
478:                 Position2 := fpuop.Zeiger ;
479:                 Copy(Ablage, fpuop.Zeile,
                      fpuop.Zeiger,
                      Position1 - Position2) ;
480:                 Argumentx := Var ;
481:                 fpuop.Zeiger := Position1 ;
482:        END ;
483:        RETURN (Argumentx # nouebg) ;
484:      END TestRegVar ;
485:
486:      PROCEDURE TestVgl() : BOOLEAN ;
487:      BEGIN
488:        fpuop.Argument4 := F ;
489:        VergleichNr :=
                IdentifiziereString(fpuop.Zeile,
                       fpuop.Zeiger, Vergleiche,
490:        ORD(fpuop.Argument4), ORD(novgl)) ;
491:        INC(fpuop.Argument4, VergleichNr) ;
492:        IF fpuop.Argument4 = novgl
493:          THEN RETURN FALSE ;
494:          ELSE WHILE fpuop.Zeile[fpuop.Zeiger] #
                      ")" DO
495:                 INC(fpuop.Zeiger) ;
496:               END ;
497:               INC(fpuop.Zeiger) ;
498:               RETURN TRUE ;
499:        END ;
500:      END TestVgl ;
501:
502:      PROCEDURE NaechsterOperand() : BOOLEAN ;
503:      BEGIN
504:        Position1 := Pos(fpuop.Zeile, Klzu,
                        fpuop.Zeiger) ;
505:        Position2 := Pos(fpuop.Zeile, Komma,
                        fpuop.Zeiger) ;
506:        IF (Position2 < Position1) AND
              (Position2 # -1)
507:          THEN fpuop.Zeiger := Position2 + 1 ;
508:               WHILE fpuop.Zeile[fpuop.Zeiger] <=
                     " " DO
509:                  INC(fpuop.Zeiger) ;
510:               END ;
511:               IF fpuop.Zeile[fpuop.Zeiger] = ")"
512:                 THEN FehlerMeldung(" Kein Operand
                        hinter dem Komma") ;
513:                      RETURN FALSE ;
514:               END ;
515:          ELSE fpuop.Zeiger := Position1 + 1 ;
516:               RETURN FALSE ;
517:        END ;
518:        RETURN TRUE ;
519:      END NaechsterOperand ;
520:
521:    BEGIN
522:      IF TestRegVar(fpuop.Argument1, fpuop.VarNum)
523:        THEN IF NOT NaechsterOperand()
524:               THEN FehlerMeldung(" Zu wenig
                                  Operanden!") ;
525:                    RETURN FALSE ;
526:             END ;
527:      ELSIF NumFilter(fpuop.Zeile[fpuop.Zeiger])
528:        THEN Position1 := Pos(fpuop.Zeile,
                        Freiraum, fpuop.Zeiger) ;
529:             Position2 := Pos(fpuop.Zeile, Komma,
                          fpuop.Zeiger) ;
530:             IF ((Position2 < Position1) AND
                    (Position2 # -1)) OR
```

→

```
          (Position1 = -1)
531:        THEN Position1 := Position2 ;
532:      END ;
533:      Position2 := Pos(fpuop.Zeile, Klzu,
                   fpuop.Zeiger) ;
534:      IF ((Position2 < Position1) AND
           (Position2 # -1)) OR
           (Position1 = -1)
535:        THEN Position1 := Position2 ;
536:      END ;
537:      Position2 := fpuop.Zeiger ;
538:      Copy(fpuop.VarNum, fpuop.Zeile,
            fpuop.Zeiger, Position1 -
            Position2) ;
539:      fpuop.Argument1 := Num ;
540:      fpuop.Zeiger := Position1 ;
541:      IF NOT NaechsterOperand()
542:        THEN FehlerMeldung(" Zu wenig
                          Operanden!") ;
543:          RETURN FALSE ;
544:      END ;
545:    ELSE IF fpuop.Zeile[fpuop.Zeiger] = Komma[0]
546:      THEN FehlerMeldung(" Kein Operand vor
                          dem Komma!") ;
547:      ELSE
          InOut.Write(fpuop.Zeile
          [fpuop.Zeiger]) ; InOut.WriteString
          ("     ") ;
          InOut.WriteCard(fpuop.Zeiger, 2) ;
          InOut.WriteLn ;
548:        FehlerMeldung(" Fehlende
                          Operanden!") ;
549:    END ;
550:    RETURN FALSE ;
551:  END ;
552:
553:  IF fpuop.Operation = tst
554:    THEN Weitermachen := TRUE ;
555:    ELSE Weitermachen :=
              TestRegVar(fpuop.Argument2,
                          fpuop.VarNum) ;
556:      IF Weitermachen
557:        THEN Weitermachen :=
                          NaechsterOperand()
558:        ELSE FehlerMeldung(" Operand nicht
                          korrekt!") ;
559:          RETURN FALSE ;
560:      END ;
561:  END ;
562:  IF Weitermachen
563:    THEN IF TestRegVar(fpuop.Argument3,
                   fpuop.Boolean)
564:      THEN IF NaechsterOperand()
565:        THEN IF NOT TestVgl()
566:          THEN FehlerMeldung
567:                ("Vergleichstyp
                nicht definiert!") ;
568:                RETURN FALSE ;
569:              END ;
570:        END ;
571:      ELSE FehlerMeldung(" Operand nicht
                          korrekt!") ;
572:            RETURN FALSE ;
573:      END ;
574:  END ;
575:  SucheEnde ;
576:  RETURN TRUE ;
577: END OperandenIdentifizieren ;
578:
579: PROCEDURE Syntaxpruefung() : BOOLEAN ;
580: BEGIN
581:   CASE fpuop.Operation OF
582:     move : IF (fpuop.Argument1 = Var) AND
                 (fpuop.Argument2 = Var)
583:            THEN FehlerMeldung(" Fmove immer
                     nur unter | Beteiligung
                     von Registern!") ;
584:              RETURN FALSE ;
585:            END |
586:     sincos : IF (fpuop.Argument2 < Reg0) OR
                  (fpuop.Argument2 > Reg7) OR
587:              (fpuop.Argument3 < Reg0) OR
                  (fpuop.Argument3 > Reg7)
588:            THEN FehlerMeldung(" Fsincos
                  arbeitet nur mit |
                  Registern als Zielen!");
589:              RETURN FALSE ;
590:            END |
591:     cmp : IF (fpuop.Argument2 < Reg0) OR
              (fpuop.Argument2 > Reg7)
592:          THEN FehlerMeldung(" Der zweite
              Operand muß | ein Register
              sein!") ;
593:            RETURN FALSE ;
594:          ELSIF fpuop.Argument3 # Var
595:          THEN FehlerMeldung(" Der dritte
              Operand muß | eine BOOLEAN-
              Variable sein!") ;
596:            RETURN FALSE ;
597:          END |
598:     tst : IF fpuop.Argument3 # Var
599:          THEN FehlerMeldung(" Der dritte
              Operand muß | eine BOOLEAN-
              Variable sein!") ;
600:            RETURN FALSE ;
601:          END |
602:   ELSE IF fpuop.Argument3 # nouebg
603:        THEN FehlerMeldung(" Die Operation
            hat | zwei Operanden!") ;
604:          RETURN FALSE ;
605:        ELSIF (fpuop.Argument2 < Reg0) OR
              (fpuop.Argument2 > Reg7)
606:        THEN FehlerMeldung(" Der Zieloperand
            muß ein | Register sein!") ;
607:          RETURN FALSE ;
608:        END ;
609:   END ;
610:   IF fpuop.Operation = movecr
611:     THEN IF (fpuop.VarNum[0] = "6") AND
             (fpuop.VarNum[1] > "3")
612:       THEN FehlerMeldung(" Der ROM-Offset
                           ist zu groß!") ;
613:         RETURN FALSE ;
614:       END ;
615:   END ;
616:   IF (fpuop.Argument1 = Num) AND
       (fpuop.Extension = E)
617:     THEN FehlerMeldung(" Zahlen haben maximal
       | Double-Precision!") ;
618:       RETURN FALSE ;
619:   END ;
620:   RETURN TRUE ;
621: END Syntaxpruefung ;
622:
623: PROCEDURE IdentifiziereString(Zeile : ARRAY OF
              CHAR ; ZeigerZeile : SHORTCARD ;
624:                        StringArray :
                      ARRAY OF SMALLSTRING ;
625:                        ZeigerAnfang,
                      ZeigerEnde : SHORTCARD)
626:                        : SHORTCARD ;
627: VAR Zeiger1, Zeiger2 : SHORTCARD ;
628: BEGIN
629:   Zeiger1 := ZeigerAnfang ;
630:   WHILE NOT (Equal(Zeile, StringArray[Zeiger1],
             ZeigerZeile, 0,
631:          Length(StringArray[Zeiger1])) OR
             (Zeiger1 = ZeigerEnde)) DO
632:     INC(Zeiger1) ;
633:   END ;
634:   IF Zeiger1 # ZeigerEnde
635:     THEN REPEAT
636:          Zeiger2 :=
           IdentifiziereString(Zeile,
           ZeigerZeile, StringArray,
637:
          Zeiger1 + 1, ZeigerEnde) ;
638:          IF Length(StringArray[Zeiger2]) >
             Length(StringArray[Zeiger1])
639:            THEN Zeiger1 := Zeiger2 ;
640:          END ;
641:        UNTIL Zeiger2 = ZeigerEnde ;
642:   END ;
643:   RETURN Zeiger1 ;
644: END IdentifiziereString ;
645:
646: PROCEDURE Equal(VAR s1, s2 : ARRAY OF CHAR; p1,
              p2, l : SHORTCARD) : BOOLEAN ;
647: VAR Laenge1, Laenge2 : SHORTCARD ;
648: BEGIN
649:   IF l = 0
650:     THEN RETURN TRUE ;  (* Immer gleich *)
651:   END ;
```

→

```
652:        Laenge1 := Length(s1) ; Laenge2 :=
                                        Length(s2) ;
653:        IF (p1 + 1 - 1 > Laenge1) OR (p2 + 1 - 1 >
                                       Laenge2)
654:          THEN RETURN FALSE ; (* Strings zu klein,
                                  Absturzgefahr! *)
655:        END ;
656:
657:        WHILE (1 > 0) AND (s1[p1+1-1] = s2[p2+1-1])
              DO
658:          1 := 1 - 1 ;
659:        END ;
660:        IF 1 = 0
661:          THEN RETURN TRUE ;
```

```
662:          ELSE RETURN FALSE ;
663:        END ;
664:      END Equal ;
665:
666:      PROCEDURE ChIsLetter(ch: CHAR) : BOOLEAN ;
667:      BEGIN
668:        IF ((ch >= "A") AND (ch <= "Z")) OR ((ch >=
              "a") AND (ch <= "z"))
669:          THEN RETURN TRUE ;
670:          ELSE RETURN FALSE ;
671:        END ;
672:      END ChIsLetter ;
673:
674:    END Parser ;
```



Wer gezielt einkauft und Public Domain Software eines bestimmten Themenbereiches bevorzugt, wird sich über die hier vorgestellten PD-Pakete besonders freuen. PD-Pool Anbieter haben die "Top Tausend" Bestenliste aus der PD-Szene Zeitschrift zum Anlaß genommen und daraus einige attraktive Programmsammlungen zusammengestellt. Weitere Pakete werden folgen. Mehr darüber lesen Sie in PD-Szene: Jetzt mit 68 Seiten, Ihr PD-Pool Anbieter hat sie.

* unverbindlich empfohlener Verkaufspreis

Top 1000 - Multipaket (5 Disks, s/w) DM 39,- *

Der richtige Einstieg: Minitext 2.3 • Letter 2.0 - zum Briefeschreiben • Filter • PD-Text 2 • Profitext • ST-Writer 1.75 • Phraser • Autocopy • Bit 3.5 • Fastcopy 3.0 • Copy_FMT • D_COPY 1.91 • Diskaux • Filecopy • HYPCOPY • M-Copy 87 • Datebook • Diskinfo • Emula 5.1 - der Mono-/Farbemulator • FREEMEM • Fileselect • Kalender • Maustreiber • Monomon • Mauspos • Retten HD • Setz_Uhr • Switcher • Protect - Virenschutz für die Festplatte • Sagrotan 4.14 - beliebter Virenkiller • Virentod • VCS-Kill • VSCOPE • Adressdatei • Adressle • ADR-2 • Adressmanager etc ...

Top 1019 - Desktop Accessories DM 39,- *

Alle ACCs aus der PD-Szene: Accload • ACC Loader 2.1 • Adress • Alarms • Bic Macc • Bit 3.5 • Calculator • Calendar • Change-ACC • Clipboard • Code • CLI • Deskuhr • Diskutil • Drupatch • DTA • Filedump • Hardcopy + Clock • Helfer • HEX-Calc • Intersect • Janine • JJ-Load • Keyboard • Key Help • Last Word • Lock • Manager • MB_Uhr • MS_DOS Step • Multi-Acc • Musik • NEC_Code P6 • Notizblock • Oki-Init • Privat Eye • Protect • RAMdisk Backup • Remember • Schieber • Screen Save • Simple • Solitaire • ST Klick • Telefon • Terminkartei • Terminplaner • TI 59 • Umlaut • Weck-Acc • Werkzeugkiste • Word 400

Top 1008 - Lernprogramme DM 49,- *

Lernpaket für Vokabeln, Mathematik, Geografie, Führerschein und einige exotische Wissensgebiete. 7 Disketten, doppelseitig formatiert und gefüllt mit den besten Programmen aus der PD-Szene: Atarus • Ausland • Brain • Bruchrechnen • CW-Trainer • ECS • Erdkunde • Fakultät • Führerschein • Genius 2 • Geograph • Globus • History • Hypervoc • Kidprogramme • Koch • Kopfrechnen • Letterator • Life • Lineare Algebra • Midi Pauk • Periode • Quicklearn • Schlange • STrans 1.2 - mit 7800 Vokabeln • Type 3 • Vokabel • Vokav plus • Wirtschaftsrechnen etc ...

Grafik - Power 1 DM 18,- *

Das Super-Paket für alle die sich mit Grafik, Design und DTP beschäftigen: Little Painter 4.32: Hat alle Features, die man zum Bearbeiten und Erstellen von Bildern benötigt. • Convert: Überträgt Bilder von einem Format in viele andere. • Printing Press V3.03: Verfügt über umfangreiche Gestaltungs- und Druckmöglichkeiten für Briefköpfe, Poster, Disklabels, Banner, Briefumschläge und Grußkarten. Arbeitet mit 9- und 24-Nadlern von Epson und NEC. • Printing Press Picturedisk 001: Diese Erweiterungsdiskette enthält viele Grußkarten und Bildblocks für Printing Press 3.03

Top 1091 - Spiele (s/w) DM 69,- *

Spielpaket für den Monochrommonitor. 10 Disketten, doppelseitig formatiert und gefüllt mit den besten Action-, Strategie-, Denk- und Geschicklichkeitsspielen aus der PD-Szene: 12. Jahrhundert • Airline Manager • Alea • Anno 1700 • Ball • Ballerburg • Billard • Collums • Doppelkopf • EAT • Empire Maps • Feuerwehr • Hotelier • Isola • Lander • Metropol • Minenfeld • Minigolf • Mirakel • Motodrom • Niemals • Northcott • Öko-Reforger • Panik • PD-Eden • Pentimo • Play Me • Ramses • Ratte • Roulette • Soko • Soliaire • Tangran • Telebörse • Tritris und noch viel mehr ...

Signum - Power 2 DM 20,- *

Enthält Utilities, Fonts und Grafiken für Signum-Anwender: SNAPFONT: Das Font-/Grafik-Interface wandelt Bilder in Fonts für 9- und 24-Nadeldrucker • Bigfont & Turnpic: Erstellt große Überschriften aus Signum-Fonts. Als Bild gespeichert, können die Headlines in vielen Programmen verwendet werden. • CHsets-Fi & Suchsets: Ermittelt sämtliche Fonts, die in einem SDO verwendet wurden. • SigShell: Für die wichtigsten Programme. • SigHelp: ACC-Leiste für Hardcopies. • Masstab & Vorlage: Bemassung in cm. • Clip-Art aus verschiedenen Themenbereichen

Top 1092 - Spiele (f) DM 69,- *

Spielpaket für den Farbmonitor. 10 Disketten, doppelseitig formatiert und gefüllt mit den besten Action-, Strategie-, Denk- und Geschicklichkeitsspielen aus der PD-Szene: Adel • Alien Blockade • Ascot • Azarian • Biker - Motorradspiel • Canfield & Klondike - Patiencen • Celestial Cesars • D.G.D.B 2 • Darts • Droid • Dungeon Master Save-Disk mit neuen Levels • Drachen • Hellfire • Imperium - sehr komplexes Strategiespiel • Queboid - für Joystickkünstler • Space War • Stone Age • Thonatos - Tron-Variante • Two-Game und mehr ...! Lassen Sie sich überraschen!

Midi - Power 3 DM 12,- *

Das Paket für alle, die auf dem ST Musik machen. 32-Track: 32-Spur Sequencer. • AMI: Über Algorithmen entsteht Musik • BWSOUND: Zum Erstellen und abspielen eigener Sounds. Mit Soundbibliothek. • D-50PD: Übertragen und Speichern von Soundbänken mit Roland D50. • DX21Over: Editor für Yamaha DX21, DX27 und DX100. • ESQUIZIT: ESQ-1 Library mit vielen Features • EZSEQ (f): Leicht bedienbarer Sequencer mit enormen Funktionen • FB-01: Editor für YAMAHA FB-01 • Freedrum 2: Schlagzeug-Synthi mit zwei Soundlibrary • MUSIC: Zum Komponieren & Notendruck.

Top 1018 - Utilities DM 39,- *

Fünf Disketten mit den besten Utilites aus der PD-Szene: 50/60 HZ Monitorumschaltung • ASCII-ED • AUTO-Selector • Autocopy • Automat • Autoverify • Bit 3.5 • BOOTKONF • Bootwähler • BOOT-DEF • CMDSVR • Datasave • Desk-Pic • Desk Switcher • DIR-Analyse • Diskaux • Disk Engineer • Disk Speed • Emula 5.1 • Filecode • Goodview: Bildschirmlupe • Hard-Inf • HD-Info • LO-MI-HI • Maboot • Maustreiber • Quick-Index • Quick ST • Recover • Rescue • Slow-Down • Speeder • Super-File-Copy • Sweep • Total Delete • TREE Pack: Archivierungsprogramm • VirusScope • XDIRLIST und noch einiges mehr...

Wissenschaft - Power 4 DM 35,- * *

Analy-ST: Funktionsplot, Ableitungen, Integralfunktionen, Flächen- und Nullstellenberechnung und Übernahme der Daten in jede Textverarbeitung • Funcalc • Plotter • Plot-it • 3D-Plot & Carpet • Diskret • Dreieck: Berechnet aus drei typischen Werten eines Dreiecks alle anderen • Körper • Matrix • Statist • Molekül • Chemie • Laborant: Identifiziert bis zu 90% aller anorganischen Verbindungen • Translator: Übersetzungshilfe mit 8000 Vokabeln • V-Crack • Wordtrainer & Voktrainer • Geograph & Welt: Erdkunde-Lernprogramme • Globus • Klima • Littleprof: Mathepauker

Alle hier vorgestellten PD-Pakete erhalten Sie bei dem PD-Pool Anbieter in Ihrer Nähe. Bitte umblättern ...



Der Trend zum Großprogramm zeichnet sich immer deutlicher ab. Seitdem es die 2000er Serie und damit die doppelseitige PD-Diskette gibt, nutzen immer mehr PD-Autoren den vorhandenen Speicherplatz bis zur Grenze aus. Da wird nicht an Grafik- und Soundqualität gespart.

Die zunehmende Professionalität der PD-Software zeigt sich auch im Spielebereich: Leveleditoren und zusätzliche Spielpläne gehören wie selbstverständlich zum Lieferumfang und geben dem Spiel besonderen Reiz. Wie stark der PD-Anwender bei dieser Entwicklung Einfluß nimmt, zeigt sich gerade in letzter Zeit: Updates und komplett überarbeitete Upgrades beliebter PD-Software gehen immer öfter auf Kundenwünsche zurück.

Mit freundlichen Grüßen,

Die "23"

2121



Die Weltraum **Odyssee** von Markus Kronenberg versetzt Sie an Bord eines Raumschiffes, dessen Steuerung infolge einer Kollision blockiert ist. Beschädigtes Frachtgut, defekte Bordcomputer und blockierte Zugänge stellen den Starship-Captain vor eine neue und interessante Aufgabe. Netter Sound, ansprechende Grafik und das neue Befehlssystem machen die Weltraum Odysee zu einem Genuß für Adventure-Freunde. Auf einen echten Parser und die damit verbundenen Verständigungsprobleme wurde zugunsten einer leicht erlernbaren Kommandostruktur verzichtet: Das Programm spricht deutsch! (f, MB).

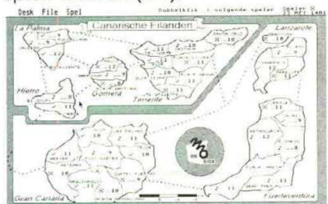
2122

AORIST wurde zum Erlernen griechischer Vokabeln entwickelt: Ein Zeichensatz mit sämtlichen Akzentkombinationen ist über die Tastatur erreichbar. AORIST verwaltet Vokabeln in mehreren Formen - z. B. Genitiv oder starker Aorist, etc. - und mit mehreren Bedeutungen in der Übersetzung. Mit dem Programm erstellte Vokabellisten können in Signum! verarbeitet werden (s/w, MB).

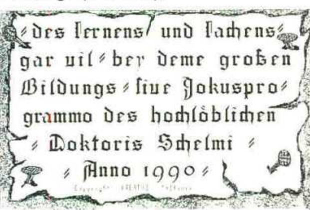


Kopfsalat hilft Probleme beim Kopfrechnen zu beseitigen. Das kleine Lernprogramm arbeitet in drei Schwierigkeitsstufen, von denen die letzte wohl nur für Rechenkünstler gedacht ist (s/w).

Memoria testet Ihr Gedächtnis: 9 Zahlenfelder werden in wechselnder Reihenfolge aktiviert und müssen per Mausklick nachgespielt werden (s/w).



Risky ist eine schöne Variante des beliebten Eroberungsspiels, von dem es sich - dank neuer Karten - deutlich abhebt. Daß der Programmautor die Kanarischen Inseln zum Schauplatz des Konflikts macht, läßt den Spieler an Touristenströme und Sandburgen denken. Weitere Karten können mit dem eingebauten Editor hergestellt werden, allerdings werden hier holländische Sprachkenntnisse vom Spieler verlangt (s/w, MB).



Allgemeinbildung will das Quizprogramm **Dr. Schelm** unters Volk bringen. Die vorliegenden Demo-Version ist im Abfrageteil auf "Alle Wissensgebiete" und "30 Fragen" begrenzt, was dem Spielspaß jedoch keinen Abbruch tut. Dr. Schelm kommentiert die Antworten mal geistreich-humorvoll, mal lehrreich (s/w, MB).

2123

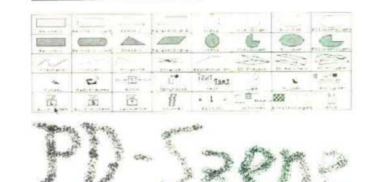
Belldrop der Lagerarbeiter soll in **Khan** mehrere Container an die dafür vorgesehenen Lagerplätze verschieben. Level für Level arbeitet er seinem Ziel entgegen: Ob es ein Ende gibt, hängt auch vom Spieler ab, der mit dem internen Editor zusätzliche Levels entwerfen kann (f, J).

Auf der **Rückkehr von XOR** wird das Raumschiff PIONEER 3 in heftige Gefechte verwickelt. Wer die Wellen feindlicher Raumschiffe meistert, muß die PIONEER nur noch durch den Asteroidengürtel steuern (f, J).

Truzzi ist scharf auf Diamanten, die er in **Zyklop** entdeckt hat. Mit jedem aufgesammelten Stein wächst die Freude und damit auch die Geschwindigkeit mit der sich unser kleiner Held auf die Suche nach weiteren Diamanten macht. Der Spieler hat dabei alle Mühe, gefährliche Hindernisse zu umgehen (f, J).

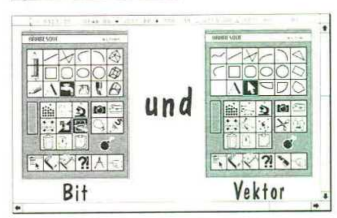
Xenotron gibt dem Spieler nur ein Leben, in dem er sich gegen zahlreiche Angriffswellen durchsetzen muß (f, J).

2124



Heiko Gemmels **PAD** versucht die Vorteile bekannter Malprogramme mit neuen Funktionen zu verbinden. Wichtigste Neuerung: PAD verarbeitet Bilder aller gebräuchlichen ST-Formate, darunter auch Farbbilder, IMG- und Blockformat (s/w).

2125



ARADEMO zeigt Ihnen die faszinierende Welt des Arabesque. Das Programm verbindet Raster- und Vektorgrafik, ist dabei aber erstaunlich einfach in der Handhabung. Arabesque läuft auf allen ATARI ST mit mindestens 1MB Speicher (viel Speicher - viel Bild), doppelseitigem Diskettenlaufwerk und SM124. Druck wird auf alles ausgeübt, was 9 - oder 24 Nadeln hat, bzw. durch eine ATARI oder HP-kompatiblen Laserstrahl geschwärzt wird (auch im Demo ist der Ausdruck möglich!). Das Bedienungskonzept geht davon aus, daß mit der linken Maustaste Funktionen ausgewählt und ausgeführt werden. Mit der rechten Maustaste werden Parametereinstellungen aufgerufen, bzw. Funktionen abgebrochen.

Besonderheiten:

f	= Läuft nur in Farbe
s/w	= Läuft nur monochrom
E90	= Läuft auch mit Emula 5.1 von Diskette 2090.
e	= in englischer Sprache
J	= Joystick notwendig
MB	= 1MB RAM erforderlich
S	= Shareware
G	= GFA-Quellcode liegt bei
K	= Kontaktkarte eingebaut

2126

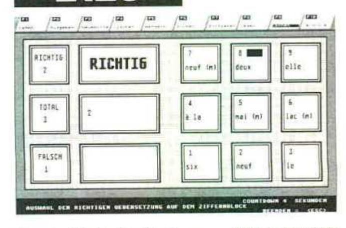
Clip-Art 11 enthält Raster- und Vektorgrafiken im PAC-, bzw. CVG-Format von Calamus.



2127

In der **Adventure-Werkstatt** werden umfangreiche Abenteuer erstellt, ohne daß dafür besondere Programmierkenntnisse erforderlich sind. Das Programm bietet viele interessante Möglichkeiten: Eigene Titelmusik und Titelbild, internes Grafik- und Soundprogramm, Aktion auf dem Bildschirm (Objekte verschwinden, Türen öffnen sich wirklich ...), Zeit- und Befehlsfallen, nähere Untersuchung von Objekten, Folgebefehle und vieles mehr (f, MB).

2128



Der Vokabeltrainer **MNEMOS** verfügt über 20 Lektionen mit Vokabeln in Deutsch, Englisch und

Französisch. Bei frei wählbarer Abfragerichtung kann beispielsweise der Übersetzungsmodus Englisch/Französisch eingestellt werden: Eine echte Herausforderung für Sprachstudenten. Umfangreiche Druckertreiber und eine Online-Hilfe sind vorhanden (s/w, MB).

2129

Drucken ist Einstellungssache: Das Accessory **setLQ500** hilft (s/w).

Farbtext schreibt auf dem Farbmonitor in mittlerer Auflösung. Neben den üblichen Textfunktionen enthält das Programm von Christina von Keitz noch sämtliche Blockoperationen, eine interne Druckeranpassungshilfe, Taschenrechner und Karteikasten (f).

Christoph Bartholme legt mit dem **Idealist 2.0** eine erheblich erweiterte Version seines Spaltendruckers vor: Menüeinträge und Parametereinstellungen können nun über die Tastatur bedient und in Form einer Anpassungsdatei gespeichert werden. Dreispaltiger Ausdruck ist möglich, ein "Nano"-Zeichensatz bringt noch mehr Text zu Papier (s/w).



IMX Liga 2.0 verwaltet Sportligen mit 20 Mannschaften und 500 Partien pro Saison, 999 Toren pro Spiel und 128 Runden pro Meisterschaft. Die Ausgabe aller Daten erfolgt auf Bildschirm, Drucker oder Diskette. IMX erstellt Toto-Tips, aktuelle Tabellen und solche von früheren Spieltagen. Statistikfunktionen und grafische Darstellungen ermöglichen eine umfassende Auswertung der Mannschaftsleistung, bis hin zur Publikation in Signum!, Calamus und ähnlichen Programmen. Das Shareware-Produkt wird ständig erweitert (MB).

Sweep 2.3 sucht und löscht unerwünschte Dateien von Diskette oder Festplatte: Das Programm ist bei der Pflege Ihres Datenbestands von großem Nutzen. Sweep wird über eine Dialogbox bedient, deren Elemente sich weitgehend selbst erklären. Die vorliegende Update-Version ist noch komfortabler geworden.

2130



Fibumat - PD 2.9 ist ein Arbeitsmittel, mit dem Klein- und Mittelbetriebe alle notwendigen Buchhaltungsaufgaben bis hin zur Bilanzierung abwickeln. Einzige Einschränkung: Maximal 1200 Buchungen pro Jahr sind möglich. Unternehmen mit einem Jahresumsatz bis etwa 500.000,- DM sollten damit ohne Probleme arbeiten können (s/w).



AktienStar II wurde für den kleineren Aktionär entwickelt, der sein Aktienpaket auf einfache und schnelle Weise verwalten möchte. Nach Eingabe der Kursdaten wird die Anlageentwicklung grafisch dargestellt: Grafiken können zur späteren Weiterverarbeitung abgespeichert werden (s/w).



Interne Anweisungen

Der Redaktion ist es gelungen, eine innerbetriebliche Mitteilung der Deutschen Bundespost zu ergattern. Sie wurde am 13.11.1989 von der Oberpostdirektion Hannover/Braunschweig unter dem Zeichen 28-2 B 3531-141/88 an sechs verschiedene Fernmeldeämter (die wir hier nicht nennen können) geschickt und trägt den Betreff "Angebliche Freigabe in bezug auf das Anschließen postalisch nicht genehmigter Telefone, Modems usw. durch das rechtskräftige Urteil des Landgerichts Hannover vom 19.10.89 gegen einen Journalisten".

Die innerbetriebliche Anweisung zeigt eindeutig, wie sich die Fernmeldeämter nach dem gültigen Gerichtsurteil des Landgerichts Hannover verhalten sollen. Dabei bezieht sich der Absender des Briefs auf den damals aufgebauschten Fall des Journalisten Kemmer, der ein Taiwan-Telefon und ein Modem angeschlossen hatte und vor dem Landgericht Hannover freigesprochen wurde. Nach Angaben der Post ist es weiterhin nicht zulässig, nicht genehmigte Modems anzuschließen: "Nach §15 Abs. 1 FAG wird bestraft, wer entgegen den Vorschriften des FAG eine Fernmeldeanlage errichtet oder betreibt. Mit der Anschließung nicht zugelassener Endeinrichtungen an das öffentliche Netz wird nach ständiger Rechtsprechung und herrschender Meinung in der Literatur eine Fernmeldeanlage unzulässig errichtet. Hieran ändert auch die mit dem Poststrukturgesetz geschaffene Liberalisierung des Fernmeldebereichs nichts, denn auch §1 Abs. 3 FAG hält ausdrücklich am Erfordernis der Zulassung fest. Dem entspricht, daß nach §168 Abs. 2 TKO Endeinrichtungen zugelassen sein müssen. Dabei ist nach derzeitiger Rechtslage auch unter Berücksichtigung des EG-Rechts nicht ausreichend, daß das Gerät im Ausland zugelassen ist. Erforderlich ist nach wie vor die nationale Zulassung durch das ZZF (§4 FZulV). (...) Eine darüber hinausgehende EG-Richtlinie, wonach die Zulassung in einem EG-Mitgliedsstaat auch in den anderen EG-Staaten Gültigkeit hat, wird zwar angestrebt, gibt es jedoch derzeit noch nicht. Auch die Entscheidung des Bundesverfassungsgerichts vom 22.06.88, wonach §15 Abs. 2a FAG für verfassungswidrig erachtet wurde, steht der Strafbarkeit der Anschließung nicht zugelassener Geräte nicht entgegen, da dieses Urteil den Straftatbestand des §15 Abs. 1 FAG unberührt läßt." Dem Inhalt dieses Briefs ist nichts mehr hinzuzusetzen, er dürfte die Rechtslage und den Stand der Post eindeutig belegen. Trotzdem weiterhin viel Spaß bei der DFÜ!

Datex-P wurde aufgefrischt

Unglaublich aber wahr: Seit dem 1.1.1990 bietet die DBP Telekom Datex P-PADs auch mit 2400 Baud an - in den Tagen der 19200 Baud-Modems wahrhaft eine technologische Wunderleistung! Durch diese Neuerung wird die DBP Telekom wieder ein kleines Stück "konkurrenz"fähiger. Die Nummern der verfügbaren PADs finden Sie in unserer Tabelle.

Weiterhin seit dem 1.1.1990 geändert wurde die Möglichkeit, sein Paßwort ohne Hilfe eines beamteten Postlers und viel Briefverkehr zu ändern. Die Prozedur geht folgendermaßen vor sich:

(1) Datex P-Rufnummer des Paßwortänderungssystems eingeben: 45010019000 oder 45020029000

daraufhin kommt die Meldung

```
Datex P: Verbindung hergestellt mit ....
```

(2) Eingangsdialog. Das System meldet sich mit

```
DATEX-P-Teilnehmerkennungssystem
der Deutschen Bundespost TELEKOM
```

und fordert den Benutzer auf, sich vorzustellen.

```
>> Bitte Teilnehmerkennung Teil A eingeben: DAAAAA
>> Bitte Teilnehmerkennung Teil B eingeben: HHHHHH
Eingabe bitte: (? gibt Ihnen eine Übersicht über die möglichen Befehle)
>>
```

Jetzt können Sie folgende Befehle eingeben:

```
1: Ändern des Paßworts
2: Informationen zur Teilnehmerkennung
3: Ende des Dialogs mit dem System
?: Hilfe
```

Alles weitere ist selbsterklärend und leicht zu durchschauen. Keep hacking!

Anschluß eines Modems

Immer noch erreichen die Redaktion Briefe, in denen der Anschluß eines Modems erfragt wird. Für alle Neugierigen hier also noch einmal eine komplette Bauanleitung zum Anschluß eines Modems an eine Zweidraht-Telefonanlage. Mitteilung für Abmahnfirmen und die DBP Telekom: Der Anschluß eines nicht zugelassenen Modems an das öffentliche Fernsprechnetz ist strafbar!

Schrauben Sie die Anschlußdose Ihres Telefons oder das Telefon selbst auf. Im Telefon sehen Sie meistens eine vorsintflutliche Platine, in der Anschlußdose zwei Kabel. Sollten Sie das Modem direkt an der Anschlußdose anschließen wollen, brauchen Sie nur die beiden Kabel, die aus der Wand kommen und zum Anschlußka-

Vermittlungsstelle Datex-P	Onkz (Vorwahl)	Anschlußnummern für Baud-Rate 300	1200	1200/75	2400
Augsburg	0821	36791	36781	36761	-
Berlin	030	240001	240081	240061	240211
Bielefeld	0521	59011	59021	59041	-
Bremen	0421	170131	14291	15077	-
Dortmund	0231	57011	52011	52081	-
Düsseldorf	0211	329318	8631	320748	134533
Essen	0201	787051	791021	793003	-
Frankfurt	069	20281	20291	20201	20251
Hamburg	040	441231	441261	441281	441291
Hannover	0511	326651	327481	327591	548181
Karlsruhe	0721	60241	60381	60581	-
Köln	0221	2911	2931	2951	2971
Mannheim	0621	409085	39941	39951	-
München	089	228730	228630	228758	299978
Nürnberg	0911	20571	20541	20501	66051
Saarbrücken	0681	810011	810031	810061	810081
Stuttgart	0711	299171	299061	299291	870321
Wiesbaden	06121	36011	36041	36081	-

Tabelle : Die Datex P-Rufnummern

bel gehen. Schrauben Sie sie ab. Die schlechtere, aber auch mögliche Methode ist, das Modem direkt an das Telefon anzuschließen. Das Anschlußkabel, das von der Anschlußdose kommt, beinhaltet vier Kabel. Von diesen Kabeln benötigen Sie das weiße und das braune. Nehmen Sie also den Plastikstecker ab und lösen Sie das weiße und braune Kabel aus der Plastikverankerung. Nun nehmen Sie das RJ11-Anschlußkabel, das mit dem Modem mitgeliefert wurde, und schneiden einen der beiden daran befestigten Stecker ab. Trennen Sie die Ummantelung ab und legen Sie die vier im Strang enthaltenen Litzen ab. Von diesen benötigen Sie nur das grüne und das rote, die anderen beiden können Sie bedenkenlos kappen, sie stören nur. Isolieren Sie die Enden der beiden Kabel soweit ab, daß sie, je nach Verfahrensweise, in die Anschlußdose bzw. in das Telefon gesteckt werden können. Nun können Sie die beiden Enden des Modemkabels mit den zuvor freigelegten Enden verbinden. Welches Kabel dabei an welches angeschlossen wird, ist gleichgültig - Zweidrahtanlagen arbeiten mit Wechselstrom.

Eine elegantere Lösung ist, sich einen Wechselstecker zu kaufen, der in vielen Kaufhäusern oder direkt bei Ihrem Modem-Händler erhältlich ist. Diesen Stekker können Sie in die Telefonbuchse stekken, auf dem Stecker selbst ist eine Buchse zum Anschluß des RJ11-Kabels. Die Vorteile liegen auf der Hand: Zum einen müssen Sie das schöne Kabel nicht zerschnippeln, Sie können das Modem auch zu einem Freund transportieren und dort anschließen, und Sie können das Modem in dringenden Fällen sehr schnell aus der Anschlußdose entfernen.

Gelockerte Bestimmungen

Ab dem 1.7. dürfen auch Telefone von Fremdherstellern an das heimische Telefonnetz angeschlossen werden. Dies wird möglich durch die Neustrukturierung der Post und die damit verbundenen Gesetzesänderungen. Trotz neuer Bestimmungen hat die DBP Telekom weiterhin gut lachen: Das anzuschließende Telefon muß weiterhin eine Zulassung des Zentralamts für Zulassungen im Fernmeldewesen haben und dürfte damit wesentlich teurer sein, als man sich das wünschen würde. Um Gerüchten vorzubeugen: Bei Modems wurden die Bestimmungen nicht gelockert. Die Regelung ab dem 1.7. bezieht sich lediglich auf Telefone. Alle DFÜ-Freaks werden leider noch eine unbestimmte Zeit warten müssen, bis sie an der Reihe sind. Genaue Aussagen waren auch auf der CeBIT nicht erhältlich. Spätestens am 1.1.1993 soll es aber endlich so weit sein, daß grundsätzlich auch Modems betrieben werden dürfen, die in einem beliebigen EG-Land bereits eine Zulassung besitzen. Darunter fallen z.B. Modems der Marke BEST, Datatronics oder Lightspeed. Trotzdem wird jedoch auch für die in anderen EG-Ländern zugelassenen Geräte weiterhin eine ZZF-Zulassung nötig sein, da nicht jeder EG-Partner die gleichen Zulassungsbestimmungen hat - zur Zulassung in Deutschland müssen dann lediglich die noch nicht getesteten Kriterien beurteilt werden.

Adapter und Filter

Welcher 2400 Baud-User hat sich noch nicht über die ärgerlichen Störzeichen geärgert, die der Gebührenzähler-Impuls der DBP Telekom verursacht (16 KHz-Signal)? Wirre Zeichen, die sich über den gesamten Bildschirm erstrecken, oder Modems, die grundlos auflegen, gehören ab sofort der Vergangenheit an. Die Kieler Firma Telekommunikation Kaben Riis (TKR) bietet ab sofort Filter für den störenden Impuls an, die einfach nur auf die Anschlußdose aufgesteckt werden müssen. An der anderen Seite des Filters befindet sich eine Buchse für die allen Modems mitgelieferten RJ11-Stecker, in die das Modemkabel nur noch eingesteckt werden muß. Der Adapter inklusive Gebührenimpulsfilter kostet DM 29,80, die Ausführung ohne Filter kostet lediglich DM 10,50.

Bezugsquelle:

Telekommunikation Kaben Riis (TKR)
Projensdorfer Straße 14
2300 Kiel
Tel. (0431) 337881

Redaktionsgeflüster

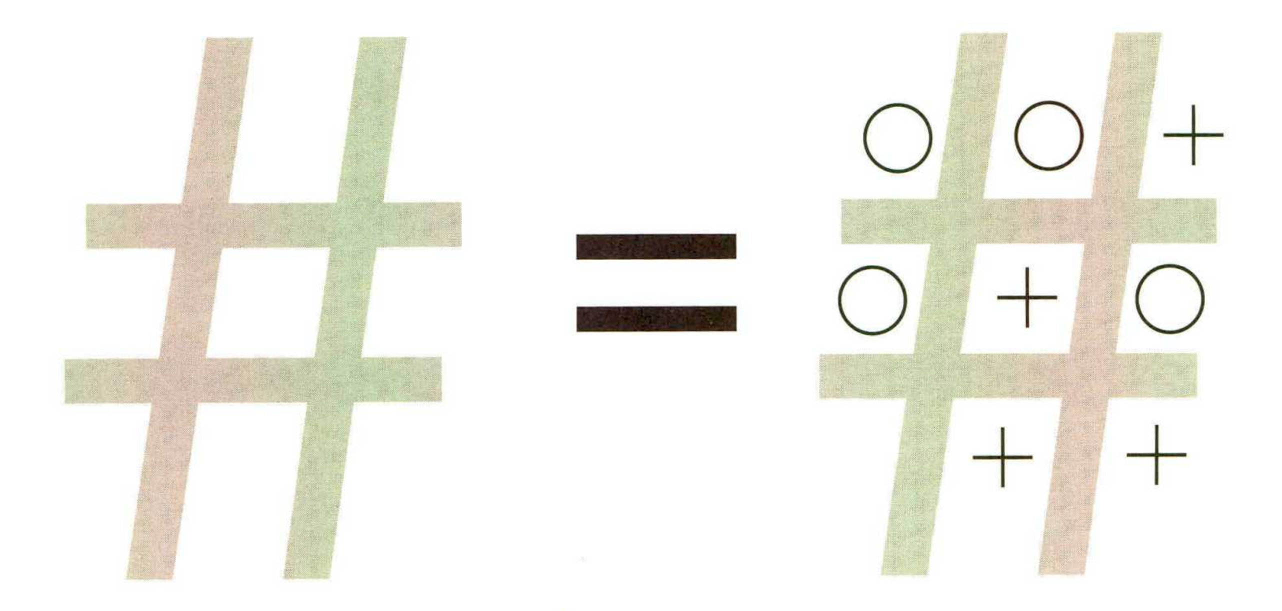
In letzter Zeit hört man immer wieder das Gerücht, die DBP Telekom werde in nächster Zeit ein Courier HST-Modem mit Zulassung für DM 60,- pro Monat anbieten. Vergessen Sie das Gerücht, es stimmt nicht: Ein Mitarbeiter der FTZ in Darmstadt beantwortete die Frage leider negativ. Ein Courier HST befindet sich weder bei der FTZ oder ZZF zum Test, noch ist die Einführung eines solchen Modems geplant.

Die Zustände bei der Post haben sich tatsächlich geändert! Kaum zu glauben, und doch wahr: Der Redaktion wird ein MDG 19k2-31 (bekannt unter dem Namen Logem T-2000 oder Trailblazer, alle drei von Kabelmetal) kostenlos zur Verfügung gestellt. Es kann von uns selbst angeschlossen werden (!) und kostet nicht (!) DM 65,- Anschlußgebühr. Herzlichen Glückwunsch zu diesem Wandel! Weitere Wandlungen: Die Mitarbeiter der FTZ Darmstadt sind freundlich und hilfsbereit. Kaum zu glauben! Sollte sich dieses Phänomen in nächster Zeit weiter durchsetzen, könnte eine Zusammenarbeit mit den Postlern auch für den privaten DFÜler recht interessant werden. Fragen Sie doch einfach bei Ihrem Fernmeldeamt nach!

Preisausschreiben : Die Gewinner stehen fest

Das Preisausschreiben in der März-Ausgabe hat eine wahre Flut von Einsendungen ausgelöst. Weit mehr als sechshundert Lösungskarten sind eingegangen; die richtige Lösung lautete "*0#", denn mit dieser Zeichenkombination gelangt man immer in das Hauptmenü des BTX-Systems der DBP Telekom zurück. Alle Gewinner wurden per Post benachrichtigt; von hier aus noch einmal herzlichen Glückwunsch!

MP



Tic Tac Toe

oder das Geheimnis des Doppelkreuzes

Wer die Vorschau der letzten Ausgabe gründlich studiert hat, dem ist mit Sicherheit die Vorankündigung unseres magischen Doppelkreuzes aufgefallen. Dort hieß es:

"Noch wollen wir Ihnen nicht verraten, was sich hinter diesem Doppelkreuz versteckt. In jedem Falle wird in der nächsten Ausgabe der ST Computer das Geheimnis des 'Gartenzauns' gelüftet werden. Lassen Sie sich überraschen."

Daß es uns dabei nicht ganz so ernst war, haben schon viele Leser vermutet. Uns ging es in erster Linie darum, den Stil so mancher anderer Zeitung zu parodieren und Ihnen, liebe Leser, zu zeigen, wie man wirkungsvoll auf folgende Ausgaben hinweisen kann. Sicherlich sind wir dabei das Risiko eingegangen, daß wir es uns mit Ihnen verscherzen, da der 1.April schon lange vorüber ist, aber wir präsentieren Ihnen auf diesen Seiten ja nicht nur die Auflösung, sondern immerhin ein komplettes Listing von Tic Tac Toe, das mit Begleittexten von Claus Brod und Meinhard Ullrich dekoriert wurde. Ich kann Ihnen garantieren, daß sich die Lektüre lohnt. Wer jemals den Film "Wargames" gesehen hat, der weiß zumindest den Wert eines Tic Tac Toe-Listings zu schätzen. Wir sind sicher, daß auch Ihnen diese vier Seiten den Spaß wert sind. Was sind schließlich vier Seiten bei fast 200 Seiten Umfang.

HE

P.S.: Sollten Sie erste Vorschläge für eine Eröffnungsbibliothek haben, schicken Sie sie an uns. Wir werden dann versuchen, sie zu implementieren.

```
/*****************************************************
** TIC TAC TOE V1.0**
** HALL OF FAME:
**    Meinhard Ullrich     12h
**    Claus Brod            2h
**
** Das Ziel dieses Spiels ist es, auf dem 3x3-
** Spielfeld drei eigene Steine horizontal, vertikal
** oder diagonal in einer Linie anzuordnen. Es wird
** abwechselnd gezogen. Wenn keiner der Kontrahenten
** einen Fehler macht, endet das Spiel zwingend
** remis! (Jaja, das gilt für viele Strategiespiele,
** aber hier ist der Rechner wirklich nicht in der
** Lage, Fehler zu machen... ist ja auch ein 68000...
** ähem)
** Das Spiel eignet sich wegen seiner Überschaubar-
** keit hervorragend dazu, die Minimax-Strategie für
** Zwei-Personen-Nullsummenspiele zu demonstrieren.
** Denn statt einer zweifelhaften und subjektiv
** vorbestimmten Bepunktung einer vorgegebenen
** Stellung kann die Bewertungsroutine für jede
** Stellung eindeutig auf Verloren, Remis bzw. Gewon-
** nen prüfen. Da es nämlich im Vergleich zu anderen
** Strategiespielen wie zum Beispiel Schach nur
** relativ wenig Knoten im gesamten Spielbaum gibt,
** kann das Minimax-Verfahren für jede Stellung
```

```
** korrekt angeben, welches Ergebnis eine Seite
** zwingend erreichen kann.
**
** Auf der Minimax-Strategie setzt hier noch das
** Alpha-Beta-Verfahren (mit tiefen Schnitten) auf,
** mit dessen Hilfe große Teile des Spielbaums als
** werden, so daß der Spielbaum beträchtlich gestutzt
** redundant erkannt werden kann. Dies führt zu einem
** erheblichen Geschwindigkeitsgewinn. Der Nachteil
** daran ist allerdings, daß somit nicht nach dem Zug
** gesucht werden kann, der dem Gegner die meisten
** Möglichkeiten läßt, Fehler zu begehen. Von mehre-
** ren Zügen, die dem Rechner zum Beispiel das Remis
** sichern, wird daher keiner favorisiert. Die
** frappierende Konsequenz: Es kann passieren, daß
** das Programm den eigenen Sieg hinausschiebt, was
** besonders gemein aussieht und vom Gegner meist
** auch so empfunden wird und werden soll.
**
** Sachdienliche Begleitliteratur bitte an jeder Po
** lizeidienststelle abzugeben. Hunde an die Leine
** führen!
** Version 0.5 (Sommer 1989): AMIGA-Version,
**                            unverschämt unschlagbar
** Version 1.0 (Mai 1990):    ST-Version, noch ohne
**                            Eröffnungsbibliothek
**                            8-), für Turbo C 2.0
** Version 2.0 (März 1997):   Erste Version für die
**                            Entwicklerexemplare
**                          ~ des TT.
*************************************************************/

/****************** Includes *********************/
# include <stdio.h>
# include <tos.h>
# include <stdlib.h>
# include <ctype.h>

/****************** Konstanten ********************/
# define LEER           0
# define RECHNER        1
# define SPIELER        2
/* Was heutzutage so alles auf einem Feld stehen darf
*/

# define MINUS_UNENDLICH -2
# define NIEDERLAGE      -1
# define UNENTSCHIEDEN   0
# define SIEG            1
# define PLUS_UNENDLICH  2
/* Werte für die Stellungsbewertung */

# define MARCELO      Gibs_uns_Roy

# define FALSE        0
# define TRUE         (!FALSE)
/* Wie wahr, wie wahr... */

# define O(x)         Cconout(x)
# define CLS              O(27);O('E');\
   printf("* TICTACTOE V1.0 - ");\
   printf("(C)1990 Meinhard Ullrich, Claus Brod *\n")
# define PRINTAT(y, x)  O(27);O('Y');O(y+32);O(x+32)
# define DELLINE
printf("\n");O(27);O('A');O(27);O('K')
/* Für die Bildschirmausgabe - nix GEM, nur TOS */

/****************** Ulkige Typen *****************/
typedef short   Brett[3][3];
typedef struct    {  short Spalte;
                     short Zeile;
                  } Zugtyp;

/****************** Globale Variablen ************/
Brett Spielfeld;
  /* Ich leiste mir hier also den Luxus, ein zweidi-
  ** mensionales Feld zu verwenden, das global ist
  ** und das ausschließlich manipuliert wird, wenn
  ** neue Stellungen erzeugt werden oder wenn Züge
  ** zurückgenommen werden. Effizienter würde das
  ** Programm arbeiten, wenn die Stellung in ein
  ** Langwort kopiert würde, das sich in einem
Register
  ** halten ließe und auf dem nur mit booleschen
  ** Operatoren gearbeitet würde. Das würde aber das
  ** Programm für den Betrachter bei weitem weniger
  ** anschaulich machen. Da sich dies zudem deshalb
  ** nicht lohnt, weil auch diese Version so schnell
  ** ist, daß sie den ersten Zug binnen weniger
  ** Sekunden findet, habe ich diese vorläufige
Fassung
  ** beibehalten.
  */
short Zugzaehler;
  /* Arbeitsbeschaffungsmaßnahme bei der
  ** Deutschen Bundesbahn  */
  /* Prototypen (nein, ich habe nicht TT gesagt) */

int   Jein(char *);
int   Wer_beginnt (void);
void  Initialisierung (void);
void  Hole_Spielerzug (Zugtyp *);
void  Zeige_Zug (short, Zugtyp *);
short Bewertung (short);
short AlphaBeta (Zugtyp *, short, short, short);
int   main (void);

/******************** Funktionen *****************/

/* Jein()
** Auf dumme Fragen gibt's auch eine dumme Antwort.
** (Goethe nicht, aber vielleicht Schiller.)
**
** Liefert TRUE (für 'j'/'J') oder
** FALSE (für 'n'/'N').
*/
int Jein(Frage)
char *Frage;
{
   char c;
   int selbsterklaerende_variable;

   printf("%s (j/n)?\n", Frage);

   do
      c = toupper((int)Cnecin());
   while ((c != 'J') && (c != 'N'));

   printf("%c\n", c);
   return (c == 'J' ? TRUE : FALSE);
}

/* Wer_beginnt()
** Prosa: Fragt den Benutzer, wer anfangen soll, und
**        liefert als Ergebnis die Konstante SPIELER
**        oder RECHNER zurück.
*/
int Wer_beginnt()
{
   CLS; PRINTAT(3,3);
   return (Jein("Wollen Sie beginnen") == TRUE ?
                           SPIELER : RECHNER);
}

/*************************************************/

/* Initialisierung()
** Prosa: Erzeugt leeres Brett mit Koordinatenbe-
**        schriftung auf dem Bildschirm. Das interne
**        Spielfeld wollen wir auch schnell löschen
**        (Datenschutz), und der Zugzaehler muß
**        entlastet werden, sagt die Eisenbahnerge-
**        werkschaft.
*/
void Initialisierung()
{
   int x,y;

   CLS;  /* Löscht den Bildschirm - schlierenfrei */

   PRINTAT(4,4); printf("   a b c ");
   PRINTAT(5,4); printf("1   . . . ");
   PRINTAT(6,4); printf("2   . . . ");
   PRINTAT(7,4); printf("3   . . . ");
   printf("\n");

   for (x=0; x<3; x++)
       for (y=0; y<3; y++)
         Spielfeld[x][y] = LEER;

   Zugzaehler = 0;
}

/*************************************************/
```

→

```
/* Hole_Spielerzug()
** Prosa: Wie der Name schon sagt: Gibt dem Spieler
**        die Möglichkeit, die Koordinaten seines ge-
**        wünschten Zugs kundzutun. Dieser wird in
**        die der aufrufenden Prozedur bekannten
**        Variablen Zug geschrieben. Zieht der
**        Spieler es vor, statt der Zugkoordinaten
**        ein Telegramm aufzugeben, so können und
**        wollen wir ihn nicht daran hindern. Vor
**        sicht: Nach 80 Zeichen wird das Telegramm
**        verschickt, notfalls nach Nirwana.
*/
void Hole_Spielerzug(Zug)
   Zugtyp * Zug;
{
   char Spalte, Zeile; /* Erlaubt: 'a'-'c'
                          bzw. '1'-'3' */
   char Puffer[80];    /* Die einfachste Lösung, aber
                       ** nicht ganz narrensicher.
                       */

   do {
      PRINTAT(10,1); DELLINE; printf("Ihr Zug: ");
      scanf("%s", Puffer);
      Spalte = Puffer[0];
      Zeile  = Puffer[1];
      }
   while(   (Spalte < 'a') || (Spalte > 'c')
         || (Zeile  < '1') || (Zeile  > '3'));

   Zug->Spalte = (short)(Spalte - 'a');
   Zug->Zeile  = (short)(Zeile - '1');
}

/
*********************************************************/

/* Zeige_Zug()
** Prosa: Der übergebene <<Zug>> wird auf dem Bild-
**        schirm dargestellt in der "Farbe" des
**        Spielers, der gerade <<AmZug>> war.
*/
void Zeige_Zug(AmZug, Zug)
   short  AmZug;
   Zugtyp *Zug;
{
   PRINTAT(5 + Zug->Zeile, 7 + 2 * Zug->Spalte);
      if (AmZug == SPIELER)
         puts("O");
      else
         puts("X");
}

/****************************************************/

/* Bewertung()
** Prosa: Die Bewertung prüft in diesem einfachen
**        Fall nur nach, ob der Spieler, der zuletzt
**        gezogen hat, drei Steine in eine Reihe
**        gebracht hat und somit das Spiel gewonnen
**        hat. Die einzige Eingabe ist daher
**        <<LetzterAmZug>>.
*/
short Bewertung(LetzterAmZug)
register short LetzterAmZug;
   /* Für C-Einsteiger: 'register short' bedeutet,
   ** daß der C-Compiler spezielle Hardwareeigen-
   ** schaften nutzt, um die Datenregister auf 13
   ** Bits zu kürzen.
   ** Die restlichen 19 Bits verwendet Motorola für
   ** interne Zwecke. Vorsicht: Höllisch inkompati-
   ** bel!
   */
{

   /* Teste auf Mühlen (ich dachte, wir sind bei
       TicTacToe?) */
   if (Spielfeld[1][1] == LetzterAmZug)
   {
      if (Spielfeld[0][0] == LetzterAmZug)
         if (Spielfeld[2][2] == LetzterAmZug)
           return(SIEG);
      if (Spielfeld[1][0] == LetzterAmZug)
         if (Spielfeld[1][2] == LetzterAmZug)
           return(SIEG);
      if (Spielfeld[2][0] == LetzterAmZug)
         if (Spielfeld[0][2] == LetzterAmZug)
           return(SIEG);
      if (Spielfeld[0][1] == LetzterAmZug)
         if (Spielfeld[2][1] == LetzterAmZug)
           return(SIEG);
   }
   if (Spielfeld[0][0] == LetzterAmZug)
   {
      if (Spielfeld[1][0] == LetzterAmZug)
         if (Spielfeld[2][0] == LetzterAmZug)
            return(SIEG);
      if (Spielfeld[0][1] == LetzterAmZug)
         if (Spielfeld[0][2] == LetzterAmZug)
            return(SIEG);
   }
   if (Spielfeld[2][2] == LetzterAmZug)
   {
      if (Spielfeld[2][1] == LetzterAmZug)
         if (Spielfeld[2][0] == LetzterAmZug)
            return(SIEG);
      if (Spielfeld[1][2] == LetzterAmZug)
         if (Spielfeld[0][2] == LetzterAmZug)
            return(SIEG);
   }
   return(UNENTSCHIEDEN);
}

/**************************************************/

/* AlphaBeta()
** Prosa: Sie befinden sich hier bei einer der
**        größten Sehenswürdigkeiten in diesem Pro-
**        gramm, eine echte Rarität. Die Wurzeln
**        dieser Prozedur reichen bis in die frühen
**        vierziger Jahre dieses Jahrhunderts zurück,
**        als John von Neumann und Oskar Morgenstern
**        das Minimax-Verfahren vorschlugen. Wem man
**        das Alpha-Beta-Verfahren zuordnen soll, ist
**        heute etwas umstritten, große Verdienste um
**        seine theoretische Erforschung erwarb sich
**        Donald E.Knuth.
**        Sie finden hier eine Implementierung vor,
**        die sich nicht nur einfacher, sondern auch
**        tiefer Schnitte bedient. Bei der Suche im
**        Spielbaum ist daher zu jeder Zeit ein
**        Suchfenster vorgegeben. Dieses besteht aus
**        einem vorläufigen Maximum und Minimum. Mit
**        fortschreitender Suche kann dieses Fenster
**        immer mehr eingeengt werden, so daß im
**        Verlauf der Suche immer mehr Äste des
**        Spielbaums als redundant erkannt und abge-
**        schnitten werden können. (Wenn die Prozedur
**        mit einem bestimmten Fenster aufgerufen
**        wurde und innerhalb der Prozedur ein
**        vorläufiger Wert ermittelt wird, der
**        außerhalb dieses Fensters liegt - man sagt:
**        die Fenstergrenzen werden verletzt -, so
**        kann die Prozedur sofort verlassen werden -
**        mit der Fenstergrenze, die verletzt wurde,
**        als Rückgabewert.)
**        Um noch einmal erheblich an Platz zu
**        sparen, habe ich hier statt der Minimax-
**        die Negmax-Variante implementiert, die
**        allerdings für den Laien schwerer zu ver-
**        stehen ist. (Einen Beweis für die Äquiva-
**        lenz der Verfahren sowie eine genauere
**        Besprechung findet man zum Beispiel
**        bei: Claus Brod, Meinhard Ullrich:
**             Programmierung des Mühlespiels auf
**             verteilter Hardware,
**        Studienarbeit an der Friedrich-Alexander-
**        Universität Erlangen-Nürnberg, Institut für
**        Mathematische Maschinen und Datenverarbei-
**        tung, Lehrstuhl für Programmier- und Dia
**        logsprachen sowie ihre Compiler.)
**    Überprüfen Sie regelmäßig Ihren Verbandkasten!
*/
short AlphaBeta(Inp_Zug, Inp_AmZug, alpha, beta)
Zugtyp * Inp_Zug;
short Inp_AmZug;
short alpha, beta;
{
   register short Zeile, Spalte;
   register short Wert;
   register short AktMaximum;
   Zugtyp Zug;
   register short LetzterAmZug;
```

→

```
   Zugzaehler++;

   /* Wir wollen die zeitraubende Prüfung, ob ein
   ** Spieler eine "Mühle" bauen konnte, nur einmal
   ** durchlaufen. Es kann ja immer nur der Spieler
   ** gewonnen haben, der zuletzt gezogen hat.
   ** Daher...
   */
   LetzterAmZug = (Inp_AmZug==SPIELER) ?
      RECHNER : SPIELER;

   /* Die nächsten beiden if's prüfen, ob bereits
   ** eine Endstellung vorliegt.
   */
   if (Bewertung(LetzterAmZug) == SIEG)
   {
      Zugzaehler—;
      return(NIEDERLAGE);
      /* Der Spieler, der als letzter gezogen hat,
      ** hat eine Gewinnstellung, also hat der
      ** Spieler, der nun dran ist, bereits verloren.
      */
   }

   if (Zugzaehler > 9)
   {
      Zugzaehler—;
      return(UNENTSCHIEDEN);
   }

   AktMaximum = alpha;

   /* Die Zuggenerierung ist in diesem Fall so
   ** einfach, daß es am günstigsten (weil am zeit-
   ** sparendsten) ist, sie gleich in die AlphaBeta-
   ** Prozedur zu integrieren:
   */
   for (Zeile = 0; Zeile <= 2; Zeile++)
      for (Spalte = 0; (Spalte <= 2) &&
           (AktMaximum<beta);
           Spalte++)
         if (Spielfeld[Spalte][Zeile] == LEER)
         {
            Spielfeld[Spalte][Zeile] = Inp_AmZug;
            Wert = -AlphaBeta(&Zug,
                   LetzterAmZug, -beta,-AktMaximum);
            if (Wert > AktMaximum)
            {
               AktMaximum = Wert;
               Inp_Zug->Zeile = Zeile;
               Inp_Zug->Spalte = Spalte;
            }
            Spielfeld[Spalte][Zeile] = LEER;
         }

   Zugzaehler—;

   return (AktMaximum);
}

/
***********************************************************/

/* main()
**           Allein das Wasser, das er haßte
**         Und das ihn jetzt nach oben stößt,
**         Nachdem Geröhr es nicht mehr faßte,
**         Hat ihm die Freude eingeflößt.
**         (aus "Der Gulli", Ballade in 24 Strophen
**          von Meinhard Ullrich. Bei mehr als
**          hundert Anfragen demnächst als Paperback
**          im gutsortierten Buchhandel.)
**
** Prosa: Hier passiert im Grunde nicht viel. Also
**        machen wir am besten auch gar nicht erst
**        große Worte darum.
**
**        Naja, also gut: Falls es dem Spieler danach
**        gelüstet, darf er den ersten Zug machen;
**        dann wird wechselweise die AlpaBeta-
**        Prozedur gestartet, um einen Zug für den
**        Rechner zu ermitteln, und der Benutzer nach
**        seinem nächsten Zug befragt. Weil ich mir
**        sicher bin, daß das Programm nicht verlie-
**        ren kann, ist eine solche Abfrage erst gar
**        nicht vorhanden. (Sollte jemand das Gegen-
**        teil beweisen können, bitte ich darum die
**        siegreiche Zugfolge an die Redaktion
**        einzusenden.) Sobald die Alpabeta-Prozedur
**        einen Gewinnweg für den Rechner gefunden
**        hat, wird dies eingeblendet. Ist der Sieg
**        für den Rechner perfekt, wird dies natür
**        lich gemeldet und das Spiel ist abrupt zu
**        Ende.
*/
int main()
{
   short Erfolgsaussicht;
   Zugtyp Zug;
   int Anfang;
   int NocheinSpiel = TRUE;
   int gewonnen;

   while (NocheinSpiel == TRUE)
   {
     Anfang = Wer_beginnt();
      Initialisierung();
      if (Anfang == SPIELER)
         {
         Hole_Spielerzug(&Zug);
         Zeige_Zug(SPIELER, &Zug);
         Spielfeld[Zug.Spalte][Zug.Zeile] = SPIELER;
         Zugzaehler++;
         }

      gewonnen = FALSE;

      while ((Zugzaehler < 9) && (gewonnen == FALSE))
      {
         /* Den vorgeschlagenen Zug legt die
         ** AlphaBeta-Prozedur in der vorgegebenen
         ** Struktur Zug ab:
         */
         Erfolgsaussicht = AlphaBeta(&Zug, RECHNER,
                                 MINUS_UNENDLICH,
PLUS_UNENDLICH);
         Zeige_Zug(RECHNER, &Zug);
         Spielfeld[Zug.Spalte][Zug.Zeile] = RECHNER;

         /* War der letzte Zug der Gewinnzug? */
         if (Bewertung(RECHNER) == SIEG)
         {
            PRINTAT(12,1); printf("Ich habe
                               gewonnen!\n");
            gewonnen = TRUE;
         }
         else
         {
            /* Ist der Sieg schon sicher? */
            if (Erfolgsaussicht == SIEG)
            {
            PRINTAT(11,1); printf("Ich werde
                               gewinnen!\n");
            }

            Zugzaehler++;
            if (Zugzaehler != 9)
            {
                 do Hole_Spielerzug(&Zug);
               while(Spielfeld[Zug.Spalte][Zug.Zeile]
                 != LEER);
                 Zeige_Zug(SPIELER, &Zug);
                 Spielfeld[Zug.Spalte][Zug.Zeile] =
                   SPIELER;
                 Zugzaehler++;
               }
           }
      }

      if (gewonnen == FALSE)
      {
         PRINTAT(12,1);
         printf("Naja... einigen wir uns auf
                 Unentschieden.\n");
      }

      PRINTAT (14,1); NocheinSpiel = Jein("Noch ein
                                         Spiel");
   }
   return 0;
}
```

Weil Qualität kein Zufall ist!

Die Grafikserie mit Stil:

von Spitzengrafikern gefertigt,
hohe Auflösung (300 DPI),
jeden Monat neue Motive,
umfangreiche Themenübersicht,
copyrightfreie Anwendung (kein PD)

"Take off"

je Ausgabe (3 Disketten) DM 34,90

ST Profi-Partner * Mönkhofer Weg 126 * 2400 Lübeck * ☎ 0451-505367 * FAX 0451-505531

UMSON ST ...

... bekommen Sie zwar außer unserer Info nichts bei uns, jedoch aktuelle und leistungsfähige Programme zu fairen Preisen!

COMPTABLE ST (Buchführung) — DM 198.00
Ein Buchführungsprogramm für Geschäft und Privat. 500 Konten (DATEV), 10 Steuersätze, bis zu 10000 Buchungen. Steuersätze und Privatanteilsätze können bereits im Kontenrahmen vorgegeben werden. Auswertungen : Saldenliste, Kontenblätter, Journal, Kassenbuch, Gewinn/Verlust-Rechnung, Umsatzsteuerdaten jeweils für einen beliebigen Monat, ein beliebiges Quartal oder das ganze Jahr! Alle Ausgaben auf Bildschirm, Drucker oder Datei. Buchungsmemo für 25 Buchungen (einfach durch Anklicken abrufbar). Universelle Druckeranpassung. Incl. 50- seitigem, reichhaltig illustriertem Handbuch (mit Ausdruckbeispielen) im Ringordner!

K-FAKT ST (Professionelles Fakturierungssystem) — DM 398.00
Adress-, Artikel und Stücklistenverwaltung, Umsatz/Absatzstatistik, frei wählbare Indexfelder, Erstellung von Angebot, Auftr.-Best., Lieferschein, Rechnung, Storno, Gutschrift, Mahnung, Anfrage, Bestellung, Eingangsbeleg. Aufmaßberechnungen. Alle Arten von Listen und Formularen, Anbindung an 1st Word Plus. Besteuerung nach Artikel oder Kunde, somit auch für Steuersystem in der Schweiz geeignet! Universeller Druckertreiber, an alle Drucker anzupassen. Incl ausführlichem, über 100 Seiten starken Handbuch mit zahlreichen Illustrationen. DEMO-VERSION DM 20,-- HANDBUCH VORAB DM 50,- (wird angerechnet !)

ST-MATHETRAINER II (Lernprogramm) — DM 59.00
Für 1.-6. Schuljahr. 1x1 und Mischrechnen mit wählbaren Höchstzahlenwerten. Umrechnung von Gewichten und Längenmaßen, im Schwierigkeitsgrad durch eingebaute Editierfunktion frei an den Lernbedarf bzw. den Schulbuchstoff anzupassen. Mit Benotung und Protokoll. Voll in GEM eingebunden! Incl. deutscher Bedienungsanleitung.

ST-RECHTSCHREIBEN II (Lernprogramm) — DM 59.00
Für 1.-6. Schuljahr. Wörter in Sätze einfügen, Singular und Plural, Kommata setzen. Im Schwierigkeitsgrad durch eingebaute Editierfunktion frei an den Lernbedarf bzw. Schulbuchstoff anzupassen. Mit Benotung und Protokoll. Voll in GEM eingebunden! Incl. deutscher Bedienungsanleitung.

TKC-VIDEO (Verwaltung von Videocassetten) — DM 79.00
Verwaltet bis zu 5000 Videofilme pro Datei. 10 Felder für Titel, Spielzeit, Zählwerk, Darsteller, "verliehen an ...", Bemerkungen, usw. . Umfangreiche Such- u. Selektierfunktionen. Ausdruck von Listen + Etiketten (frei über ASCII-Maske zu gestalten). Voll unter GEM, deutsche Bedienungsanleitung.

TKC-MUSICBOX (Musiktitelverwaltung) — DM 79.00
Verwaltet bis zu 5000 CDs, LPs oder MCs. Suche nach Einzeltiteln, Audruck von Haupttitelübersicht u. Gesamttitelliste mit MC-Nummern und Etiketten. Voll unter GEM, incl. deutscher Bedienungsanleitung.

TKC-TRAINER (Trainingsprogramm) — DM 99.00
Sehr hoher Lernerfolg durch Karteikastenprinzip. Für Sprachen, Chemie, Physik, etc. Voll unter GEM, leicht auch für Schüler zu bedienen. Incl. deutschem Handbuch.

TKC-BANKMANAGER (mit Sammler-Ausdruck) — DM 99.00

ST-GIRO PLUS — DM 49.00
Überweisungsdruckprogramme für den geschäftlichen (häufigen) und privaten Gebrauch. Auch für Schecks + Lastschriften. Ausdruck über ASCII-Masken an alle Formulare anzupassen!

ST-VOKABELTRAINER (Trainingsprogramm) — DM 49.00
Vokabeltraining für Englisch, Französisch, Italienisch, Spanisch und zeichenkompatible Sprachen. 100 Vokabeln pro Datei, voll unter GEM

TKC-TERMIN/ADRESS (Terminkalender) — DM 99.00
Universeller Terminplaner mit integrierter Adressverwaltung. Erstellung von Terminlisten, Adresslisten und Serienbriefen. Voll unter GEM, incl. ausführlicher Bedienungsanleitung!

TK COMPUTER-TECHNIK Thomas Kaschadt
Hard- & Software - Entwicklung & Vertrieb

Bischofsheimer Straße 17 * Postfach 60
D-6097 Trebur-Astheim
Fernruf : (06147) 3550
FAX 06147-3555
Btx. 06147-3555

24-Stunden-Auftragsannahme per Anrufbeantworter!

Bei uns werben bringt

GEWINN

Sprechen Sie mit uns.
Heim Verlag ☎ 06151/56057

Über 1000 PD-Disks für den ATARI ST

Hier nun unser TOP-Angebot!!!

Für nur 30,- DM (Scheck/bar) je Paket erhalten Sie portofrei 5 2dd-Disketten gefüllt mit Super-PD-Software incl. unseren 80-seitigen POWER-Katalog!!! (Ausland 35,- DM je Paket!)

- ☆ Paket 1: Enthält die besten, spektakulärsten **PD-Spiele** (s/w).
- ☆ Paket 2: Enthält die besten **Anwenderprogramme!** (s/w)
- ☆ Paket 3: Ebenso spektakuläre **PD-Spiele** aber in **Farbe!**
- ☆ Paket 4: **Einsteigerpaket** mit sehr nützlichen Programmen s/w.
- ☆ Paket 5: Enthält hunderte von **Clip-Art-Bildern**. Super!! (s/w)
- ☆ Paket 6: Enthält die besten **Musik- & MIDI-Prg.** (1MB+s/w).
- ☆ Paket 7: **Hot-Girls**: Erotikpaket nur für Erwachsene (1MB+s/w)
- ☆ Paket 8: Enthält fantastische **SUPER-Demos** in Farbe (1 MB).

Die "Special-Pakete":

12 Pakete mit ausgewählter Software. Das Beste vom Besten!!! Für nur 20,- DM (bar/Scheck) je Paket erhalten Sie versandkostenfrei auf 2 2dd-Disketten unser "Special-Paket" incl. 80-seitigen Hauptkatalog (Ausland 25,- DM/Paket).

Brettspiele (s/w) Monopoly, Schach, uvm.	**Ballerspiele (s/w):** Arkanoid-Variante, uvm.	**Textverarbeitung:** ... sehr leistungsstark (s/w)!
Glücksspiele (s/w): Einarmige Banditen u.a.	**Denkspiele (s/w):** Shanghai-Variante, Tetris, uvm.	**Datenbanken (s/w):** ... frei Maskenerstellung...
Farbspiele (f): TOP-Spiele für Farmonitor.	**Signum Utilities:** Hilfsprogramme und Fonts.	**Utilities (s/w):** Sehr empfehlenswert!
Zeichenprogramme: CAD, Malprg. ... (s/w)	**Musik (s/w):** 2 ganz starke Programme.	**Erotik (s/w+1 MB):** Nur für Erwachsene! Alter!

Hier nun weitere Angebote:

- G Copy II 84,90
- Signum II .... a. A.
- Calamus Fonteditor .. 189,-
- STAD V1.3+ 169,-
- Anti Viren Kit II 84,90
- F16-Falcon (dt.) 71,90
- Arabesque 248,-
- Populous 71,90
- Indiana Jones Adv. 69,90
- Dungeon Master, dt. .. 72,90
- Great Court 72,90
- Italy '90 63,90
- North & South 63,90
- Pirates 63,90
- Rainbow Warrior 63,90
- Zak Mc Kracken 73,90
- American Dreams 63,90
- Black Tiger 52,90
- Flight Command 74,90
- Hyperaction 52,90
- Soundmaschine II 179,-
- T.I.M. V1.2 259,90
- Adimens Plus V3.0. 329,-
- Mega Paint II 448,-
- Kick Off 44,90
- Chaos Strikes Back 62,90
- Rainbow Island 52,90
- Midwinter 78,90
- Drakken 71,90
- Jack Niclas Golf 63,90
- Leisure Suit Larry II 74,90
- Precious Metal 67,90
- Rainbow Island 52,90
- Xenomorph 67,90
- Wayne Gretzky Hock. 63,90
- Aquanaut 63,90
- Conqueror 72,90
- Full Metal Planet 63,90
- Manchester United 52,90

Vesand: 5,- DM bei Vorauskasse oder 7,- DM per Nachnahme.

Computer-Software ☆ Ralf Markert
☆ Balbachtalstr. 71 ☆ 6970 Lauda 1 ☆ ☎ 09343 / 3854 ☆



20 Jahre Vorsprung in der Entwicklung
Seit 1967-mit allem, was zur OOP gehört: Kapselung, Vererbung, dynamische und statische Bindung. Und noch einiges mehr: Scopes, Blöcke, Class Nesting, dyn. Strings und Arrays-alles stimmt, nichts fehlt: Eine runde Sache.

Aktive Objekte
Objekte nicht nur passiv, sondern auch mit Eigenleben-eröffnen die ganze Welt der Simulation und unzählige neue Möglichkeiten.

Safety First
Compilieren statt Interpretieren- jede Menge Prüfungen, schon vor der Ausführung: Indices, Schnittstellen, Objektqualifikationen. Dazu sicheres Initialisieren und Entfernen von Objekten durch Garbage Collector und vieles mehr.

Mainframe-Qualität auf dem Schreibtisch
Ob Desktop- oder Großrechner: Simula ist Simula. Überragende Portierbarkeit durch strikte Standardisierung von Anfang an, in jahrzehntelanger Praxis bewährte Implementierungen. Getrennte Übersetzung für Teamarbeit. Compiler und Linker erzeugen auslieferbare Programme.

Apollo DN3000
Apple MacIntosh/MPW
Atari ST
BESM 6 (UdSSR)
CDC 3000
CDC 6000/Cyber
Cromemco CS-x00
DEC 10/20
DG Eclipse/Desktop
DG MV
EC 1040/1055 (DDR)
ES EVM (UdSSR)
Honeywell Bull DPS8
HP 900/300, HP9300
IBM mainframe
IBM PC/DOS
IBM PC/OS2
IBM PC/XENIX(UNIX)
ICL 2900, ICL PERQ
Indata IN4200
Masscomp
NCR Minitower
NCR Tower XP/32
Nixdorf Targon/31
Norsk Data NORD 100
Norsk Data NORD 500
Pr1me
SIEMENS 7500/BS2000
SIEMENS 7800/BS3000
SIEMENS MX
SIEMENS PCD
Sperry 1100
SUN-3
SUN-4 (SPARC)
VAX/UNIX
VAX/VMS

Mehr dazu finden Sie in c't, Ausgabe April 90, Seite 334

PC (MS-DOS) .... 1) 448,- (298,-)
interakt. symb.Laufzeitdebugger, FP-Arithmetik nach IEEE per Software
Atari ST .... 1) 198,- (148,-)
GEM-Desktop,TOS-,AES- und VDI-Libs deutsche Dokumentation
Apple MACII .... Schutzgebühr 15,-
zur Ausführung unter MPW
PC-386(UNIX) single user .... ca. 7700,- (4600,-)
wie MS-DOS-Version, zus. Coprozessorunterstützung für Intel 80387 und Weitek x167, Wartungsabk. möglich
Sun SPARCstation1, 1J. Updateservice, interakt. symb. Laufzeitdebugger .... 2) ca. 5300,- (2800,-)

Alle Preise incl. MwSt, zzgl. Versandkosten, Preise in Klammern: Ausbildungspreis für Lehrinstitute, Studenten, Schüler u.s.w.
1) gültig bis Juli 90 einschl. 2) gültig bis Mai 90 einschl.

Alle Simula-Systeme können jetzt auch in Deutschland direkt bezogen werden über:

simulateam GmbH iG
Ref: c 06 - Postf.50 01 63 D-4600 Dortmund 50
...does it with class!

---

iks

SM 124 Multisync

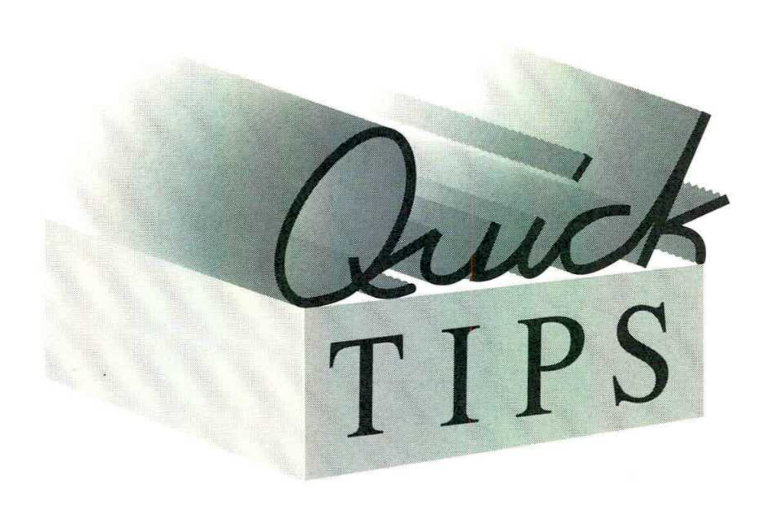
Bausatz zum Umrüsten Ihres Monochrommonitors zur Darstellung der 3 Auflösungen.
Platine incl. Kabel und allen Bauteilen **DM 129,00**
Platine fertig bestückt **DM 149,00**
Umbau Ihres Monitors **DM 249,00**
7 Tage Rückgaberecht für die Bausätze. Preise incl. Testbild und Malprogramm auf Diskette.
Bei Bestellung bitte Baujahr angeben.

Public Domain

Serie "ST Comp." ab 6 Stück DM 5,00
bis 5 Stück DM 6,00
Wir liefern bis PD 223/224 immer 2 PD's auf einer Disk. (ungerade beginnend und aufeinanderfolgend)
Ausführliche Liste und Zeitschrift PD-Szene gegen Rückporto DM 1,50.

Preise zzgl. Porto und Verpackung DM 5,00, NN plus DM 2,00, besser V-Scheck (Ausland nur V-Scheck).

iks
Schönblickstraße 7
7516 Karlsbad 4
ab 18 Uhr 07202/6793



"Einschaltverzögerung"

Wenn man stolzer Besitzer einer Festplatte ist, läuft das Booten normalerweise so ab:

1. Einschalten von Festplatte, Bildschirm, Drucker...
2. Warten, bis sich die Platte initialisiert hat
3. Einschalten des Rechners

Lästig ist eigentlich nur, daß man nicht mit einem Fußtritt auf die Steckdose alles einschalten kann, da man sonst den Reset-Knopf drücken muß, damit der ST die Festplatte erkennt (beim 1040 ist das nur lästig, aber beim Mega ST geht das gehörig auf die Wirbelsäule). Abhilfe schafft eine Einschaltverzögerung, die aber etwas teuer ist (ab 50,- DM aufwärts), oder ein Kondensator, der in den Rechner eingelötet wird (mit zwei linken Händen sollte man das lieber lassen). Die Lösung ist ganz einfach: Der Griff zum Reset-Knopf wird simuliert. Wenn der ST bootet, wartet er auf die Platte. Hat er sie in einer bestimmten Zeit nicht gefunden, geht er davon aus, daß es keine Harddisk gibt. Also wird von Diskette gebootet. Rein zufällig steht dort unser Programm. Es wartet 15 Sekunden (nun sollte die Festplatte startklar sein) und führt einen Reset durch. Die Zeilen 9 bis 14 kann man weglassen, was aber stört, falls man das Programm aus Versehen angeklickt hat.

Stefan Endrejat

```
 1: 'IS_HD_DA.PRG von Stefan Endrejat
 2: '(c) 1990 MAXON Computer GmbH
 3: '
 4: test%=BIOS(10)            !Bitmap der Laufwerke
 5: FOR i|=2 TO 15            !Teste Laufwerk C...P
 6:   IF BTST(test%,i|)=TRUE    !Wenn Bit gesetzt
 7:     SYSTEM                !Platte da, also Ende
 8:   ENDIF                   !
 9: NEXT i|                   !nächstes Bit
10: '
11: PRINT "Festplatte nicht initialisiert!"
12: zeile|=CRSLIN
13: FOR zeit|=15 DOWNTO 1     !Verzögerung b. Reset
14:   PRINT AT(1,zeile|);"Reset in ";zeit;
                       " Sekunden.";SPC(2)
15:   PAUSE 50                !1 Sekunde Pause
16: NEXT zeit|
17: '
18: ' RESET-Routine aus dem Scheibenkleister
19: '
20: ~GEMDOS(&H20,L:0)
21: LPOKE &H420,0             !Wert für Kaltstart
22: LPOKE &H426,0             !Wert für Kaltstart
23: LPOKE &H43A,0             !Wert für Kaltstart
24: reset_vektor%=LPEEK(&HFC0004)
25: CALL reset_vektor%        !byebye...
```

Listing : Kaltstart nach 15 Sekunden...

Doppelklick-Wiederholung

Wer viel mit Massenspeichern wie Festplatten oder Diskettenlaufwerken arbeitet, kennt sicher das Problem, daß manchmal bis zu 3 Ordner geöffnet werden müssen, um ein bestimmtes Programm zu laden. Dazu sind dann insgesamt 5 Doppelklicks notwendig. Das TOS erlaubt es, einen Doppelklick zu wiederholen. Klicken Sie zum Beispiel zweimal kurz auf ein Laufwerkssymbol und lassen die linke Maustaste gedrückt. Das Fenster öffnet sich trotzdem. Bewegen Sie den Mauszeiger vom Laufwerkssymbol weg, um nicht ein weiteres Fenster zu öffnen. Dann können Sie den Mauspfeil auf die zu ladende Datei bzw. den zu öffnenden Ordner bewegen (linke Maustaste immer noch gedrückt lassen). Die Datei (oder der Ordner) öffnet sich, ein weiterer Doppelklick wird nicht erforderlich. Zum Öffnen einer Datei muß die Maustaste wieder losgelassen werden. Solange die linke Maustaste gedrückt ist, müssen Sie aufpassen, daß kein falsches Objekt angewählt wird. Dies erfordert etwas Übung.

Thomas Mokler

Prozessor-Stop

Die MC68000 MPU hat an Pin 17 (-Halt) einen Anschluß, der bei allen STs unbelegt ist. Erhält dieser Pin ein Low-Signal, wird der laufende Buszyklus beendet, und anschließend geht der Prozessor in den HALT-Zustand. In diesem Zustand sind alle Adreß- und Datenleitungen hochohmig. Der Prozessor ist dann abgeschaltet. Man kann also in seinen ST einen Schalter einbauen, mit dem man GND und -Halt verbindet. Wenn dann bei Ihrem Lieblingsspiel das Telefon klingelt, können Sie den Computer anhalten und nachher wieder absturzfrei weiterspielen. Bedingung hierfür ist aber, daß während dieser Zeit keine Tastatur- oder Mauseingaben gemacht werden, weil der Tastaturprozessor nicht angehalten wird. Eine zweite Anwendungsmöglichkeit eines solchen Schalters ergibt sich für Programmierer, die fremde Programme analysieren wollen. Schaltet man z.B. bei NEOCHROME auf Halt, verändert sich der Bildschirminhalt, weil in diesem Programm Rasterzeilen-Interrupts angewendet werden. Dies erlaubt einem erfahrenen Programmierer Rückschlüsse auf das Programm. Die dritte Anwendungsmöglichkeit besteht darin, statt eines Schalters einen Schlüsselschalter einzubauen. Es muß beim Kauf darauf geachtet werden, daß sich der Schlüssel auch in geschlossenem Zustand abziehen läßt (das ist leider nicht immer der Fall). So bleibt Ihr wertvoller ST vor unberechtigtem Zugriff verschont. Wird der Rechner bei geschlossenem Haltschalter eingeschaltet, bleibt der Bildschirm schwarz; nichts funktioniert. Besitzer eines Mega ST können den -Halt-Anschluß an Pin 33 des Mega-Busses abgreifen. Der Masseanschluß ist an Pin 31 oder 32 vorhanden.

Thomas Mokler

HP DeskJet Plus

Mit Interesse habe ich als Besitzer eines DeskJet Plus den Test in der April-Ausgabe gelesen und kann dem Autor nur zustimmen. Das größte Manko, die sehr hohen Betriebskosten, läßt sich jedoch mit einem einfachen Trick drastisch senken: Da die Lebensdauer des Druckkopfes im allgemeinen die Reichweite des Tintenvorrates deutlich überschreitet, bietet sich ein Wiederbefüllen durch das Lüftungsloch an. Mit einer Injektionsspritze und einem ordinären Tintenfaß läßt sich die Patrone wieder auffüllen. Zwar wird diese Methode von HP nicht empfohlen, hat sich aber bei vielen Nutzern als erfolgreich erwiesen (siehe USENET). Der Druckkopf, der durch eine Tinte mit anderen Eigenschaften verstopfen könnte, würde ja eh weggeworfen. Ich benutze schwarze Tinte von Markenherstellern, die in Deutschland meist vom selben Lieferanten versorgt werden. Sollte dann doch eine Düse verstopfen und auch durch Reinigen nicht frei werden, verbleiben ja noch 49 andere, die Qualität bleibt auch dann noch auf höherem Niveau als bei manchem Nadeldrucker. ATARI-User sollten sich vor dem Erwerb von Hard- oder Softfont-Modulen unbedingt ansehen, ob sie mit Script oder Signum! nicht viel mehr für viel weniger erhalten.

Frank Helferich

Genaue Uhrzeit

Wenn man die Uhrzeit messen will, sind die "normalen" Pascal-Funktionen (clock, gettime) unbrauchbar, da sie beide im 2-Sekundentakt arbeiten. In der ST-Computer 7/88 (und korrigiert 1/89) wurde aber ein kleines GFA-BASIC-Programm abgedruckt, das Zeitmessung im Millisekundenbereich erlaubt. Mit der Zusatzbibliothek PASTIX ist auch die Übernahme von Peeks und Pokes (obwohl im Standard-Pascal nicht vorgesehen) ein Kinderspiel, da die betreffenden Funktionen alle vordefiniert sind. Die Umsetzung schien jedoch unmöglich, da laut Handbuch die Verwendung der XBIOS-Routine 31 nicht möglich ist. Die Praxis zeigt jedoch, daß - entgegen der Meinung von CCD - die Funktion nutzbar ist; die Uhr läuft also korrekt. Man muß nur darauf achten, daß man die Compiler-Option *$I+* nicht verwendet, sonst produziert man drei Bomben. Abhilfe für diesen Fall: die Integer-Variablen durch short-Integer und trunc durch long-trunc ersetzen. Das Listing muß mit *$i* eingebunden werden, ein erstmaliger Aufruf erfolge mit inituhr (um die Uhr zu starten), danach kann die Zeit - ähnlich wie mit "clock" - seit dem Aufruf mit uhr200 abgefragt werden - nur etwas genauer.

Philipp Maier

```
 1: procedure inituhr (var uhr: integer);
 2:   var bild: integer;
 3:       zeit: integer;
 4:
 5:   function physbase: integer;
 6:     xbios(2);
 7:   procedure xbtimer(timer,kontrolle,daten:
                 integer; routine: integer);
 8:     xbios(31);
 9:
10:   (* Einbinden der PASTRIX-Routinen *)
11:   procedure lpoke(adresse: integer;
                          wert: integer);
12:     external;
13:   procedure wpoke(adresse: integer;
                          wert: integer);
14:     external;
15:
16: BEGIN
17:   bild:=physbase;
18:   zeit:=bild+32200;
19:   uhr:=zeit+20;
20:   lpoke(zeit,146341893);
21:   lpoke(zeit+4,16775695);
22:   lpoke(zeit+8,41743359);
23:   wpoke(zeit+12,21177);
24:   lpoke(zeit+14,uhr);
25:   wpoke(zeit+18,20083);
26:   lpoke(uhr,0);
27:   xbtimer(0,2,246,zeit);
28: end;
29:
30: function zeit200 (uhr: integer): integer;
31:   var u: integer;
32:
33:   (* Noch eine PASTRIX-Funktion *)
34:   function lpeek(adresse: integer): integer;
35:     external;
36:
37: begin
38:   u       :=lpeek(uhr);
39:   u       :=u+u div 1024;
40:   u       :=trunc(u+0.5);
41:   zeit200 :=u;
42: END;
```

Listing : Eine genaue Uhr unter Pascal

Wordplus im Alltag

Zu 1st_Wordplus 3.15 ist viel Kluges geschrieben worden. Allerdings enthält das Programm seine Tücken, die besonders Umsteiger von Version 2.02 rasend machen können. Es handelt sich durchweg um Kleinigkeiten, die in Tests und Rezensionen nicht angesprochen wurden. Der Reihe nach:

Erstnutzer lesen auf Seite 50 unter Rubrik 4.2.5. "Lineale ändern", daß sie das Dialogfenster durch Anklicken des Paragraphenzeichens am linken Rand der Linealzeile erreichen. Richtiger muß es heißen "durch Doppelklick...", da Altnutzer diesen Passus sicher nicht im Handbuch lesen, sind sie auf eigene Versuchsreihen angewiesen.

Auf Seite 56 will das Handbuch Erstnutzern einreden, daß sie feste Leerzeichen nur über Mausklick ins Schlüsselkästchen, also überaus umständlich, setzen könnten. Ihnen wie auch verunsicherten Altnutzern sei gesagt: natürlich lassen sich feste Leerzeichen nach wie vor auch über die Tastenkombination "Control-Space" setzen.

Die Möglichkeit, die aktuelle Datei ohne vorheriges Abspeichern auszudrucken, ist prima. Leider wird verschwiegen, daß der Ausdruck dann mehr als doppelt so lange dauert. Wenn also auf der Diskette noch genügend Platz ist, kann es günstiger sein, doch zu speichern, um anschließend doppelt so schnell drucken zu lassen.

Altnutzer haben auch ohne deutlichen Hinweis im Handbuch sicher inzwischen erfahren, daß die Menüpunkte "Lineal bzw. Position anzeigen" mittlerweile gleichzeitig verwirklicht werden können. Wenn eine Möglichkeit mit dem Häkchen markiert wird, schaltet sich die andere Darstellungsart nicht mehr automatisch aus.

Wirklich ärgerlich ist aber nur die nirgendwo angesprochene Falle für Altnutzer über die Formatdateien auf Seite 61 des Handbuchs. Alle Umsteiger von Version 2.02, die die Möglichkeit nutzen, über bestimmte Extensions Briefköpfe, Seitenformate, wiederkehrende Texte usw. einzulesen, können über zwei Dinge stolpern, die einen rasend machen können: Wer sich wundert, wieso sein wunderschöner Briefkopf nicht eingelesen wird, obwohl das bei der Vorversion immer geklappt hat, resigniert entweder oder stellt fest, daß der Ordner für diese Dateien nicht mehr "FORMATS" heißt, sondern ganz heimlich "FORMATE". Kaum entdeckt, schon geändert - und klappt immer noch nicht? Dann arbeiten Sie mit zwei Laufwerken (Programmdisk in A, Datendisk in B) - oder mit vergleichbarer Verteilung auf zwei Partitionen der Festplatte - und haben den Ordner "FORMATE" auf

Laufwerk B und nicht auf A, wie es jetzt nötig ist. Im Ergebnis ist es so zwar besser, nur - es muß halt gesagt werden; denn welcher Routinier liest bei einem Update immer alles von vorne bis hinten durch!

Hans-Jürgen Hageman

Ermittlung der TOS-Version

Für viele Anwendungen ist es von Interesse zu erfahren, welche TOS-Version auf dem Rechner installiert und von wann sie ist (Erstellungsdatum). Mit einem kleinen GFA-BASIC-Programm, das leicht in andere Sprachen zu übertragen ist, läßt sich das Problem lösen.

HE

```
 1: ' TOS-Version und -Datum
 2: '
 3: ' Anfangsadresse des Betriebssystem-Headers
 4: '
 5: adresse=(LPEEK(&H4F2))
 6: '
 7: ' Versionsnummer des TOS im BCD-Format
 8: '
 9: version$=HEX$(DPEEK(adresse+2))
10: PRINT "TOS-Version: "+LEFT$(version$,1)+"."
         +RIGHT$(version$,1)
11: '
12: ' TOS-Erstellungsdatum im BCD-Format
13: '
14: datum$=HEX$(LPEEK(adresse+24))
15: PRINT "Erstellungsdatum: "+MID$(datum$,LEN(
         datum$)-5,2)+"."+MID$(datum$,1,LEN(
         datum$)-6)+"."+RIGHT$(datum$,4)
16: END
```

Listing : So findet man TOS-Version und -Datum.

Bild 1: Pinbelegung der ROM-Sockel im ST und des EPROMs 27010

27010	Sockel	27010	Mega-ST-ROM-Sockel	Mega-ST-ROM-Sockel	27010	Sockel	27010
Vpp		1			32		+5V
A16		2			31		PGM
A15	A15	3	1	28	30	+5V	----
A12	A12	4	2	27	29	A14	A14
A7	A7	5	3	26	28	A13	A13
A6	A6	6	4	25	27	A8	A8
A5	A5	7	5	24	26	A9	A9
A4	A4	8	6	23	25	A11	A11
A3	A3	9	7	22	24	A16	$\overline{OE}$
A2	A2	10	8	21	23	A10	A10
A1	A1	11	9	20	22	$\overline{CE}$	$\overline{CE}$
A0	A0	12	10	19	21	D7	D7
D0	D0	13	11	18	20	D6	D6
D1	D1	14	12	17	19	D5	D5
D2	D2	15	13	16	18	D4	D4
Gnd	Gnd	16	14	15	17	D3	D3

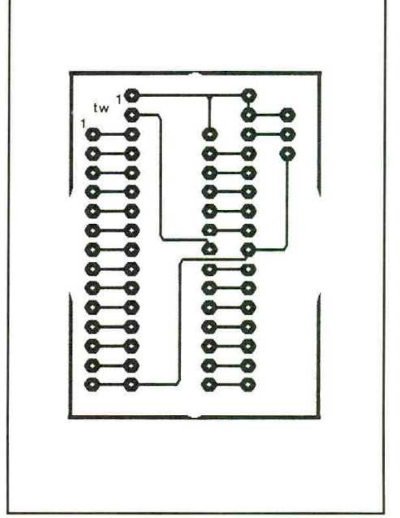
TOS, TOS Hurra

Gepatchtes TOS im Mega ohne viel Aufwand

Wenn Mega ST-Besitzer ein gepatchtes TOS verwenden wollen, waren sie bisher im Nachteil. Bei den meisten Megas (z.T. auch beim 1040) befindet sich das TOS in zwei Custom-Chips von ATARI. Da es keine pin-gleichen EPROMS gibt, mußte man entweder mit dem Original-ATARI-TOS arbeiten, oder seinen Rechner umrüsten (siehe ST-Computer 1/90 "Patch As Patch Can"). Meist sind die übrigen vier Plätze für EPROMs nicht einmal gesockelt, so daß man auch diese nachträglich einbauen muß. Dieses Projekt verhilft nun allen Mega ST-Besitzern, ohne daß sie an Ihrem Rechner herumlöten müssen, zur Möglichkeit ein gepatchtes TOS zu verwenden.

Will man das Betriebssystem in zwei EPROMs packen, werden Chips mit 128 kByte Speicherkapazität benötigt. Diese gibt es, mit ungebanktem (linearem) Adreßraum als Typ 27010. Leider besitzt dieser IC 32 Pins. Die Fassungen im ST sind aber nur für 28polige Bausteine vorgesehen. Diese Tatsache ist jedoch nicht weiter tragisch, vor allem, wenn man sich die Pinbelegung der Sockel und der EPROMs anschaut (Bild 1). Wie man sieht haben viele Pins die gleiche Bedeutung, so daß man die EPROMs fast direkt in die Sockel stecken könnte. Aber eben nur fast, ohne ein klein wenig Tüftelei kommt man nicht aus.

Bild 2: Platinenlayout



Wenn man keine Möglichkeit hat, das abgebildetet Layout (Bild 2) auf eine Platine zu übertragen, kann man sich auch mit einer Lochrasterplatine behelfen. Die meisten Verbindungen sind einfach nur durchzuschleifen. Das Pin /OE vom 27010 muß mit Masse, Bit 16 der Adresse an das richtige Pin und die Stromversorgungsleitungen müssen extra gelegt werden. Wenn Sie beim Aufbau als Sockel für den 32poligen 27010 nur Pinreihen nehmen und für den Anschluß im Mega ST Pinreihen mit langen Beinchen verwenden, können Sie durch Umstecken der Chips (TOS-ROM und 27010) das Original-Betriebssystem neben dem gepatchten verwenden.

Ein Anmerkung noch zum Aufbau der Platine. Im Mega ist recht wenig Platz zwischen den ROMs. Sie sollten deshalb die Länge der Platine so gering wie möglich halten. Für den Austausch benötigen Sie zweimal die gleiche Platine (für jedes TOS-ROM eine).

Sie werden sich jetzt sicherlich fragen, für was die drei einzelnen Lötpunkte gedacht sind. Wenn Sie hier einen Umschalter anschließen und als EPROM-Typ einen 27020-Chip benutzen, können Sie ebenso wie bei der 6-EPROM-Variante zwei Betriebssystemversionen in den EPROMs unterbringen. Der Umschalter dient zum Wechseln zwischen den Versionen (natürlich nur im ausgeschalteten Zustand des Rechners, es sei denn, Sie lieben den Anblick von Bomben auf dem Monitor des STs).

Zum Brennen der Mega-Bit-EPROMs (27010 bis 27040) bietet sich unser Junior Prommer mit dem neuem Mega-Modul (siehe ST-Computer EXTRA Nr. 3) an. Damit können Sie die neuen 32poligen EPROM-Typen genauso elegant bearbeiten, wie die herkömmlichen 28poligen Chips.

TW

Externe 16 MHz für das HD-Laufwerk

Uns haben mehrere Briefe zu diesem Thema erreicht, so daß wir dies hier noch einmal aufgreifen möchten. Wenn der 16 MHz-Takt für den Floppy-Controller, aus welchen Gründen auch immer, extern erzeugt werden muß, ist die Verwendung eines Quarzoszillators wohl die einfachste und eleganteste Lösung. Nun sollte auch der 8 MHz-Takt für den 'normalen' Betrieb des Controllers aus diesem abgeleitet werden. Das Prinzip dazu ist in Bild 1 zu sehen. Das vom Quarzoszillator kommende Signal wird auf den Takteingang eines JK-Flip-Flops gegeben. Seine Ausgänge wechseln nun, bei der dargestellten Belegung der Eingänge [J und K liegen fest auf 1 (+5V)], bei jeder fallenden Flanke des Taktsignals ihren Zustand. So erhält man am Ausgang ein Signal mit 8 MHz. Die Realisierung mit ICs ist in Bild 2 wiedergegeben. Für das Flip-Flop eignet sich sowohl der 74LS73 als auch der 74LS103. Da diese beiden Bausteine eine unterschiedliche Pin-Belegung haben, sind beide angegeben (in Klammern die für den 74LS 103).

Übrigens eignen sich für den Umbau nur die Controller mit

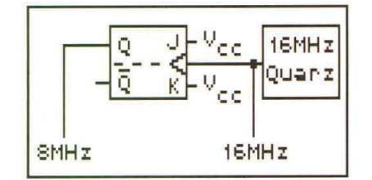


Bild 1. Prinzip der Takthalbierung

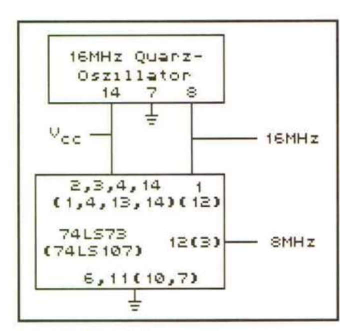


Bild 2. Realisierung des Schaltungsteils

der Bezeichnung WD1727/0202. Die alte Version WD1772/0002 arbeitet nur bis etwa 14 MHz einwandfrei.

TW

Zeichensätze in GFA-BASIC

Schon in mehreren Zeitschriften - auch in der ST-Computer - wurden Routinen abgedruckt, die die Einbindung eines eigenen Fonts ins Betriebssystem oder ein eigenes Programm erlauben. Leider waren die meisten dieser Routinen recht lang, da dort der Font oft ins ganze System integriert wird. Falls man aber nur einen Font in ein eigenes Programm einbinden möchte, gibt es eine ganz legale Möglichkeit, dies in wenigen Programmzeilen zu erledigen. Mit

```
 1:   FILESELECT  "\*.fnt","",a$
 2: IF  a$<>""  AND  EXIST(a$)
 3:       buffer%=MALLOC(4096*2)      !reservieren
 4:     s%=buffer%                    !Quellbereich
 5:      d%=buffer%+4096              !Zielbereich
 6:      BLOAD  a$,buffer%            !...und  einladen
 7:    FOR  z1&=0  TO  255
 8:       FOR  z2&=0  TO  15
 9:          BYTE{d%+z1&+z2&*256}=BYTE{s%}
10:          INC  s%
11:        NEXT  z2&
12:    NEXT  z1&
13:       FILESELECT  "\*.inl","",a$
14:     IF  a$<>""
15:          BSAVE  a$,buffer%+4096,4096
16:    ENDIF
17:     ~MFREE(buffer%)
18: ENDIF
```

Listing 2: Umwandlung von STAD-Fonts ins ATARI-Format

dem Line A-Aufruf holt man sich den Pointer auf eine Tabelle, die wiederum Pointer auf verschiedene Fonts enthält. Der zweite Eintrag enthält einen Pointer auf den 8x8-Systemzeichensatz. Dessen Eintrag mit dem Offset 84 zeigt auf den nächsten Fontheader (16x8-Systemfont). Dort findet man den Pointer auf die eigentlichen Font-Daten. Man muß sich jetzt nur noch den alten Pointer merken und kann seinen eigenen einbinden. Der Font muß aller-

```
 1: PROCEDURE  init_font
 2:    ' (c)  MAXON  Computer  GmbH  1990
 3:    '
 4:  '          LINEA_0         MOVE.L  A1,D0  RTS
 5:     linea0$=MKI$(&HA000)+MKI$(&H2009)+
                MKI$(&H4E75)
 6:       linea0%=V:linea0$
 7:         fonttab%=C:linea0%()      !Font-Tabelle   holen
 8:         fhadr1%={fonttab%+4}      !erster   Fontheader
 9:       fhadr2%={fhadr1%+84}      !Zeiger  ->  16x8-ZSatz
10:     INLINE  fadr%,4096          !eigenen  Font  laden
11:        oldfontptr%={fhadr2%+76}      !alter   Zeiger
12: RETURN
13: PROCEDURE  remove_font
14:      IF  oldfontptr%<>0
15:         LPOKE  fhadr%+76,oldfontptr%
16:    ENDIF
17: RETURN
```

Listing 1: So wird der Font eingebunden.



Haben auch Sie einen *Quick-Tip*?

Standen Sie auch einmal vor einem kleinen, aber schier unlösbarem Problem? Dann, durch Zufall bekamen Sie einen Tip und schon war es gelöst.

Ähnlich haben wir uns diese Rubrik in der ST Computer vorgestellt. Aufgerufen sind auch Sie, liebe Leser(innen)! Geben Sie Ihre Erfahrungen weiter, egal, ob es um Anwendungen, Programmieren o.ä. geht.

Wir sammeln Ihre (und unsere) Tips und stellen Sie ggf. in den Quick-Tips vor.

Beteiligen Sie sich an der ST Computer!

Einsendungen an: MAXON Computer
ST Computer Redaktion
Stichwort: Quick-Tip
Industriestr. 26
6236 Eschborn

dings im ATARI-Format vorliegen. Hier sind die Bytes so organisiert, daß zuerst das erste Byte des ersten Zeichens, dann das erste Byte des zweiten Zeichens usw. bis zum ersten Byte des 256. Zeichens folgen, dann kommt das zweite Byte des ersten Zeichens, das zweite Byte des zweiten usw. So geht es bis zum letzten Byte des letzten Zeichens weiter. Die am meisten verbreiteten Fonts sind jedoch leider im "STAD-Format", wo die Bytes für die Zeichen direkt aufeinanderfolgen. Mit Listing 2 kann man sich einen solchen Font ins ATARI-Format umwandeln.

Harald Plontke

1. Zuerst eine Reaktion auf den Quick-Tip von Thomas Appel im Heft 4/1990. Es geht dabei um die RSC-Dateien und merkwürdige Werte, die nach einem Doppelklick zurückgeliefert werden. Dies ist kein Bug, sondern ein dokumentiertes Feature des AES! Wird ein TOUCHEXIT-Objekt per Doppelklick verlassen, wird Bit 15 im Rückgabeparameter gesetzt. Das eigentliche Objekt läßt sich einfach feststellen:

```
1:  ret&=FORM_DO(tree%,0)
2:    dclick!=BTST(ret&,15)              !  Vielleicht  einen...
3:  ret&=BCLR(ret&,15)                   !  ...Doppelklick
4:    OB_STATE(tree%,ret&)=BCLR(OB_STATE(tree%,ret&),0)
```

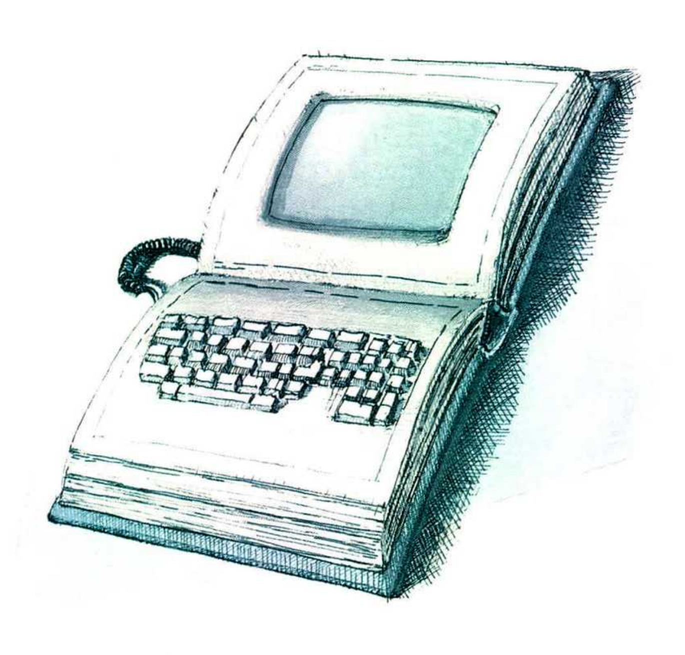
Listing 3: Verbesserung des Listings in Ausgabe 4

Dummerweise ist dieses Feature nicht im GFA-Handbuch beschrieben, jedenfalls nicht in der holländischen Übersetzung.

2. Um den Programmablauf zu steuern, fragte ich in einem GFA 3.5-Programm den Ereignispuffer sowie die Tastatur per EVNT_MULTI() ab. Die relevanten Elemente des Puffers hatte ich per ABSOLUTE gesetzt, um die Auswertung übersichtlicher zu gestalten. Im Interpreter funktionierte es perfekt, aber in der compilierten Version reagierte das Programm nur noch auf Tastendrücke: Ereignisse wurden gar nicht ausgewertet. Es stellte sich heraus, daß die Ursache des Problems darin lag, daß ich das Puffer-Array in einer Prozedur dimensionierte. Offensichtlich wanderte das Array durch den Speicher bei Garbage-Collections, wodurch die ABSOLUTE-Werte ihre Gültigkeit verloren. Lösung: Die Dimensionierung gleich am Anfang im "Hauptprogramm" und nicht in einer Prozedur durchführen.

3. In demselben Programm kam es später zu mysteriösen Abstürzen, die nur eins gemeinsam hatten: Es geschah immer während einer WIND_... Funktion. Leider war es immer an einer anderen Stelle, und die Ursache war nur schwer zu finden. Wieder einmal war die Garbage-Collection schuld an der Sache. Die Fenstertitel hatte ich, natürlicherweise, in einem String-Array aufbewahrt. Der dynamischen String-Abhandlung des GFA-BASICs zufolge wanderte auch dieses Array durch den Speicher, auch wenn es am Anfang des Programms deklariert wurde. Abhilfe schafft hier nur eins: die Titel- und Info-Zeilen Ihrer Fenster dort abzulegen, wo Sie sicher sind, daß sie auch dort bleiben. Dies funktioniert zum Beispiel per MALLOC oder INLINE, wobei Sie den String in diesen Puffer kopieren. Ist a% die Adresse des Puffers und t$ der String, dann geht es am einfachsten mit BMOVE a%,V:t$,LEN(t$). Die Adresse a% übergeben Sie dann an WIND_SET(), wenn Sie die Titel- oder Info-Zeile setzen. Eventuell könnten Sie, wenn Sie auf Nummer sicher gehen wollen, noch einen CHR$(0) an den String anhängen.

Peter Kocourek



D. und H. Kraus

Das große CALAMUS-Buch

Düsseldorf 1989
DATA BECKER GmbH
392 Seiten
DM 39,-
ISBN 3-89011-346-X

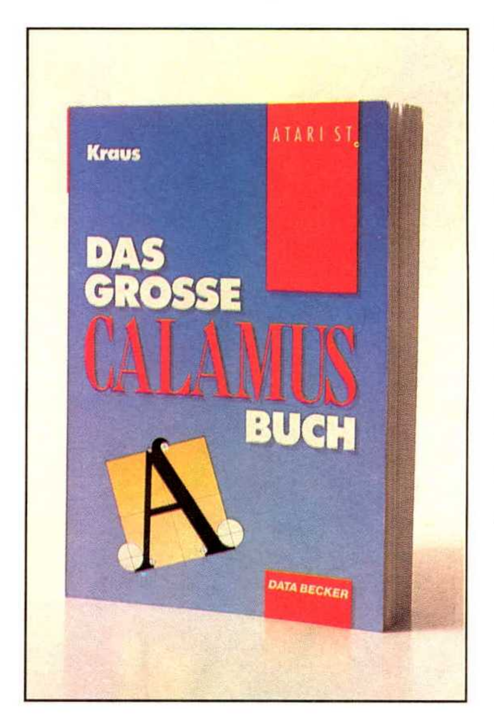
Es ist schon eine ehrenvolle Aufgabe, dem geneigten Leser eines Buches zu beweisen, daß er mit einem bestimmten Computerprogramm ganz tolle Anwendungen vollführen kann. Besonders das junge Gebiet "DTP" (muß ich die Abkürzung noch erklären?) bedarf hinreichender Erläuterung. Endlich können wir auch für dieses Marktsegment (gemeint ist DTP) eine rundum profimäßige Ausstattung vorzeigen. Im speziellen liegt das Programm Calamus der Anwenderschar und auch der ATARI Computer GmbH (siehe sogen. DTP-Center) sehr am Herzen. So sollen die komplexen Arbeitsabläufe beim Erstellen von Druckwerken mit Calamus vereinfacht, vereinheitlicht und sogar automatisierbar sein (Zitat Einleitung). Ob das Programm diesem Anspruch aber gerecht wird, muß der Leser für sich entscheiden - der Autor jedenfalls hat sich bereits entschieden (auch Zitat Einleitung).

Ich sehe solche markigen Sprüche in Vorworten mit sehr gemischten Gefühlen.

Bei dem vorliegenden Werk habe ich mir die Freiheit genommen, jenes mit dem Originalhandbuch zu vergleichen. Es ist schon erstaunlich wie sich zweieiige Zwillinge gleichen (besonders wenn sie von zwei verschiedenen Elternpaaren großgezogen wurden). Während das Originalhandbuch in seinem Teil "A" mit der Theorie ("Von Pixeln und Vektoren", "Tippen und Klikken" und "Alles im Rahmen") beginnt, startet das Große Calamus-Buch mit banaleren Dingen ("Installation", "Benutzerführung" und "Mindesteinstellungen"). Auffallend ist auch die Übereinstimmung in manchen Punkten des Layouts beider Druckwerke: Piktogramme auf jeder Seite links, deswegen auch großer Seitenabstand von links bis zum Textblock. Auch die Reihenfolge der einzelnen Unterkapitel zeigt verblüffende Ähnlichkeit. Ganz abgesehen davon ist vermerkt, daß zumindest das große Calamus-Buch mit Calamus selbst entworfen wurde.

Das große Calamus-Buch hat aber noch weitere Besonderheiten auf Lager. Bei der Beschreibung der Auswahlpunkte in den Menüs und der Icons beginnen die Kapitel sehr oft mit derselben Formulierung, wie: "Mit dieser Funktion kann ...", "Mit diesem Menübefehl wird ..." oder "Anklikken dieses Icons markiert ...". Hier hätte sich der Autor besser eine grundsätzlich andere Vorgehensweise zur Beschreibung einfallen lassen können. Das stupide Wiederholen gleicher Satzanfänge und an einigen Stellen sogar ganzer Absätze (bis auf ein paar wenige Wörter) ist nicht der begehrte Schreib- bzw. Lesestil.

Bis zum Kapitel 10 sind knapp 300 Seiten nur mit der Aufzählung der einzelnen Menü-, Icon- und Masken-Merkmale aufgefüllt. Bis dahin ist eigentlich noch nichts richtig passiert! (Da macht leider das Originalhandbuch auch keine Ausnahme.) Ab einer Stelle, wo es interessant werden könnte, bekommt der Leser nur noch Kleinigkeiten.



So gesehen in Kapitel 11 (Calamus Optionen) und in 12 (Praxisteil). Gerade hinter den 'Optionen' verbergen sich ungeahnte Manipulations-, Konstruktions- oder Gestaltungsmöglichkeiten, die leider sehr knapp und sparsam abgehandelt sind. Im sogenannten 'Praxisteil' wird durchaus die Konstruktion einer Visitenkarte ausführlich und nachvollziehbar beschrieben (einziger Lichtblick), aber von den danach abgebildeten Briefbögen fehlt jede Vorgehensweise. (Vielleicht arbeitet der Autor bereits an einem eigenen "großen" Praxisbuch zu Calamus?) Nur dieses eine Beispiel Visitenkarte ist mir für einen Praxisteil zu wenig.

Zum Schluß: In der Hauptsache besteht das große Calamus-Buch aus dem Hause DATA BECKER aus einer schrittweisen Beschreibung der einzelnen Funktionselemente - genauso wie auch das Originalhandbuch. Ausführliche Erläuterung der Theorie von Pixeln und Vektoren, dem Warum und Wie, fehlt leider. Auch der Praxisteil ist nicht sehr umfangreich. Lobenswert darf das Layout hervorgehoben werden (Calamus sei dank!) sowie die durchgängig ausreichende Bebilderung, auch mit den sehr markanten Piktogrammen. Der Preis von 39 DM kann angesichts der 392 Seiten gefüllter Information als nicht überteuert betrachtet werden.

FAZIT: Für alle, die ihr Originalhandbuch verlegt oder verliehen haben, stellt das "Große" aus dem Hause DATA BECKER einen gleichwertigen Ersatz dar.

DK

Rufus, faß!

Ideal zum (An-)Rufen

Terminal-Programme gibt es viele. Ob Rufus als Shareware-Programm den nötigen Biß hat, sich gegen die professionelle Konkurrenz durchzusetzen, wird sich zeigen.

Rufus entstand aus dem Wunsch heraus, endlich einmal ein Terminal-Programm zu besitzen, das den Anwender rundum zufrieden macht. Das scheint dem Programmierer Michael Bernards bestens gelungen zu sein. Was kann Rufus, was andere nicht können? Viel. Oder besser gesagt: Sehr viel. Rufus arbeitet unter GDOS (das AMC-GDOS von Arnd Beißner wird mitgeliefert), da es eigene Fonts verwendet. Dadurch können auch auf einem normalen monochromen Monitor mit einer Auflösung von 640x400 Punkten wesentlich mehr als 80x25 Zeichen dargestellt werden. Verschiedene Fonts für die unterschiedlichsten Bildschirmgrößen und Wünsche der Anwender werden bereits mitgeliefert, natürlich können beliebige nachgeladen werden. Sie können es sich bereits denken: Rufus läuft problemlos auf Großbildschirmen und jeder Grafikkarte, die einen VDI-Treiber besitzt. Damit müßte es auch ohne Probleme auf dem TT arbeiten. Weitere Features, die nicht unbedingt in jedem Programm eingebaut sind, erfreuen das Herz jedes Anwenders: Julian Reschkes fliegende Dialogboxen, die frei auf dem Bildschirm verschoben und größtenteils auch über die Tastatur bedient werden können, sind eingebaut und erleichtern den Umgang mit dem Programm erheblich. Ausnahmslos alle Einträge der Menüleisten können auch durch eine Kombination mit der Alternate-Taste erreicht werden - diese Nachricht dürfte Tastenakrobaten erfreuen. Wer die Maus mag, aber den langen Weg bis zur Menüleiste scheut, kann Rufus ebenfalls benutzen: Durch einen kurzen Druck auf die rechte Maustaste erscheint die Menüleiste in Form eines Pop-Up-Menüs direkt neben dem Mauszeiger. Natürlich läßt sich auch dieses Pop-Up-Menü verschieben. Damit genug zum allgemeinen Bedienungskomfort, von dem sich der Großteil der auf dem ST veröffentlichten Programme einige dicke Scheiben abschneiden kann.

Freie Auswahl

Dateien können in üblicher Manier gesendet und empfangen werden. Dazu stehen als große Auswahl die Übertragungsprotokolle XModem Checksum, XModem CRC, YModem (Batch), YModem-G (für MNP5-Verbindungen), ZModem und Stream zur Verfügung. Die beste Neuerung zu den meisten bisher bekannten Terminal-Programmen stellt ZModem dar, denn ein komfortableres Protokoll steht kaum zur Verfügung. Auch YModem-G ist in den meisten Programmen nicht zu finden und findet in Rufus einen guten Platz. Stream ist ein neues Übertragungsprotokoll, das hier zuerst in Rufus eine Anwendung findet. Stream sendet keinerlei Checksummen und ist daher nur für MNP5-Verbindungen anwendbar. Ich halte nicht viel von der Einführung eines neuen Protokolls, da es schon genug gibt, die auch bisher (noch) für alle Anwendungen reichen. Schön ist jedoch, daß die üblichen Formate inklusive ZModem verwirklicht wurden. Bei Y- und ZModem werden normalerweise die Dateinamen vom Sender vorgegeben. Möchte man dieses Feature ausschalten, existiert dazu ein Button. Diesen Vorteil konnte bisher noch kein anderes Terminal-Programm bieten. Bei ZModem existiert ein entscheidender Vorteil, den ich bisher nur von Telix auf dem PC kannte: ZModem Auto-Download. Immer wenn die angerufene Station eine Datei per ZModem senden will, registriert Rufus das automatisch und startet einen Download. Schließlich lassen sich auch Texte im ASCII-Modus senden, doch welches Programm kann das nicht? Leider existiert kein Button und kein Menüeintrag, der einen ASCII-Download starten würde. Zwar ist es möglich, den gesamten Text mitzuprotokollieren, wenn das Protokoll aber ausgeschaltet ist und man einen einzelnen Textteil abspeichern möchte, ist das nicht ohne weiteres möglich. Die einzige Möglichkeit, einen Text ASCII downzuloaden, besteht darin, den Puffer kurzzeitig zu öffnen und nach erfolgtem Download wieder zu schließen. Dazu sind allerdings mehr Tastendrücke notwendig als bei einem Programm-Download.

Umfangreiche Einstellungen für die RS232-Schnittstelle dürften allen Ansprüchen kompromißlos genügen: von 50 bis 19200 Baud läßt sich alles einstellen, was das Herz begehrt. An Emulationen bietet Rufus als eines der ersten Programme eine vollständige, komplette und

```
Datei
 INF öffnen...            ⎇L
 INF sichern...           ⎇I
 ------------------------------
 Programm starten...      ⎇X
 Shell-Aufruf             ⎇V
 Batch starten...         ⎇B
 Befehl eingeben         ⇧⎇B
 ------------------------------
 Beenden                  ⎇Q

Transfer
 Datei senden...          ⎇S
 Datei empfangen...       ⎇E
 ASCII-Datei senden...    ⎇A
 Puffer laden...         ⇧⎇L
 Puffer speichern ...     ⎇G

Block
 Block sichern...          ⎇J
 Block senden             ⇧⎇S
 -------------------------------
 Kopieren aufs Klemmbrett  ⎇C

Parameter
 RS232          ⎇R
 Terminal       ⎇T
 Protokoll      ⎇P
 Modem          ⎇M
 F'tasten       ⎇F
 Nummern        ⎇W
 Puffer        ⇧⎇P
 Zeichensatz    ⎇Z
 Pfade          ⎇D
```

100%ig funktionierende VT100-Emulation. Dabei werden auch viele Steuercodes der VT2xx-Emulationen übersetzt, so daß Rufus auch für kommende Mailbox-Standards gerüstet ist. Als weitere Möglichkeit lassen sich VT52 und TTY (ohne Steuerzeichen) einstellen. Das Echo kann auf Voll- oder Halbduplex konfiguriert und zusätzlich auch ein lokales Echo eingeschaltet werden (wichtig etwa für einige Anwendungen mit Datex-P, bei denen die Kosten für übertragene Blöcke zu hoch werden würden). Wenn Sie auch schon des öfteren gestört hat, daß hin und wieder Zeilen länger sind als 80 Zeichen und sich dazu noch alle überflüssigen Zeichen am rechten Bildschirmrand "stauen", können Sie bei Rufus einen Zeichenumbruch einstellen, der es in sich hat: Es lassen sich entweder einzelne Zeichen oder ganze Wörter umbrechen, wie es auch bei den meisten Text-Editoren üblich ist.

Riesenfenster

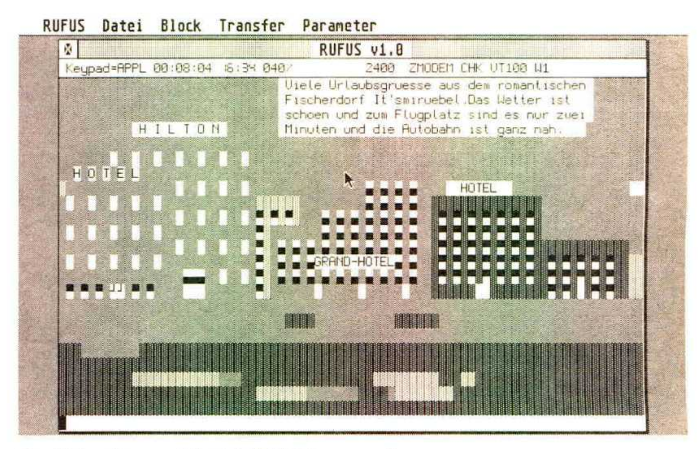
Endlich ist es auch mit einem "kleinen" Monitor möglich, mehr als 80x25 Zeichen darzustellen. Durch die nachladbaren Fonts besteht beispielsweise die Möglichkeit, 132x50 oder gar noch mehr Zeichen darzustellen. Läßt sich dann auch bei der angerufenen Mailbox die Bildschirmgröße einstellen, sind Ihrer Kreativität keine Grenzen mehr gesetzt! Das von Rufus angezeigte Fenster paßt sich automatisch der eingestellten Größe an. Ankommende Carriage Return-Zeichen können automatisch umgewandelt werden in Carriage Return plus Linefeed. Wichtig ist auch die Möglichkeit einzustellen, ob ein Backspace (ASCII 8) löschen oder lediglich den Cursor um eine Position nach links bewegen soll. Einige Boxen verwenden zwar die Einstellung 7E1 oder 7N1, senden aber trotzdem mit den Parametern 8N1. Dadurch können eventuell Probleme bei der Umwandlung der Zeichen eintreten. Deshalb kann man bei Rufus einstellen, daß auf jeden Fall alle acht Bits der eintreffenden Zeichen umgewandelt werden sollen. Bei vielen Boxen, die auf einem C64 mit dem Programm 64Sysop (oder einer Abart desselben) von Dirk Gaziç arbeiten, wird diese Einstellung eine große Erleichterung sein.

Haben Sie jemals ein original VT100-Terminal gesehen? Der Hersteller Digital verwirklichte für diese Maschinen damals den heute überall gebräuchlichen Code VT100, der von fast allen Mailboxen bedient wird. Zu einem original VT100-Gerät gehören normalerweise auch Tasten im sogenannten Application-Modus und vier Funktionstasten mit der Bezeichnung "PF1" bis "PF4". Obwohl diese Tasten heute kaum noch benötigt werden, bietet Rufus sie wie selbstverständlich an.

Auch das Modem kann konfiguriert werden; leider konnte man das bisher nicht bei jedem Terminal machen - Rufus bietet die Möglichkeit an. Die Antworten des Modems für drei verschiedene Fehler (wie "NO DIALTONE", "ERROR" oder "BUSY") können ebenso eingegeben werden wie Befehle zum Resetten ("ATZ") und Wählen ("ATDP"). Dazu gehört ein Telefonnummernverzeichnis, in das alle wichtigen Nummern eingetragen werden können.

Kommandosprache

Wer immer in den gleichen Mailboxen anruft (und das sollte wohl meistens der Fall sein), kann sich auch eigene Batch-Files für seine Lieblings-Mailboxen zusammenstellen. Soll heißen: Rufus enthält eine Kommandosprache, mit der man Logins programmieren kann. Mit den Befehlen kann man wählen, Strings senden (auch solche, die auf Funktionstasten gelegt wurden und dadurch schnell änderbar sind), empfangen, auf bestimmte Eingaben warten und vieles mehr. Fast alles, was man während einer Sitzung auch über die Menüs erreichen kann, läßt sich auch über die Kommandosprache erreichen. Genau das richtige also für DFÜ-Freaks, die täglich mit dem Medium Datenfernübertragung arbeiten. Es werden bereits mehrere Login-Dateien mitgeliefert, um sich in die verschiedensten Arten von Mailbox-Programmen einwählen zu können, etwa für Maus-, Zerberus- oder MagicBox-Mailboxen. Auch hier also ein kompletter Lieferumfang.



Endlich: eine perfekte VT100-Steuerung!

Oben erwähnte ich bereits den Puffer. Natürlich kann man auch in den Puffermodus umschalten, dazu muß lediglich die rechte Maustaste gedrückt werden. In diesem Modus lassen sich z.B. einzelne Teile des Blocks abspeichern, einziges Manko ist, daß in der Version 1.0 (noch) nicht angezeigt wird, wo der Block markiert wurde. Ist ein Emulationsmodus eingeschaltet, also VT52 oder VT100, dann werden die Steuerzeichen nicht mit in den Puffer übernommen. Der Vorteil liegt auf der Hand: Der mitprotokollierte Text kann später wesentlich leichter bearbeitet werden, da nicht alle zehn Zeilen wieder ein Steuerzeichen auftritt, das den Text zerschneidet.

Woher?

Alle Funktionen von Rufus ausführlich zu beschreiben, würde den Rahmen des Artikels bei weitem sprengen. Rufus ist ein Terminal-Programm für jede Art von Anwendung. Eine Anleitung wird ausdruckfertig ausgeliefert oder kann für DM 10,- vom Autor bezogen werden. Zu allem Überfluß kommt noch, daß Rufus Shareware ist und nur DM 50,- kostet. Allerdings sollte man dem Autor wirklich den Gefallen erweisen, das Geld zu überweisen - sofern man mit Rufus arbeitet. Viele Arbeitsstunden stecken im Programm. Sehen Sie sich Rufus am besten selbst an, es kann in vielen Mailboxen downgeloadet werden, unter anderem in der Maus Bonn (0228-254020), Maus Münster (0251-80386), Megamail (02203-83108), PEC (069-683584) (alle 300 bis 2400 Baud, 8N1, 24 Stunden) und vielen anderen Mailboxen - fragen Sie Ihren Sysop.

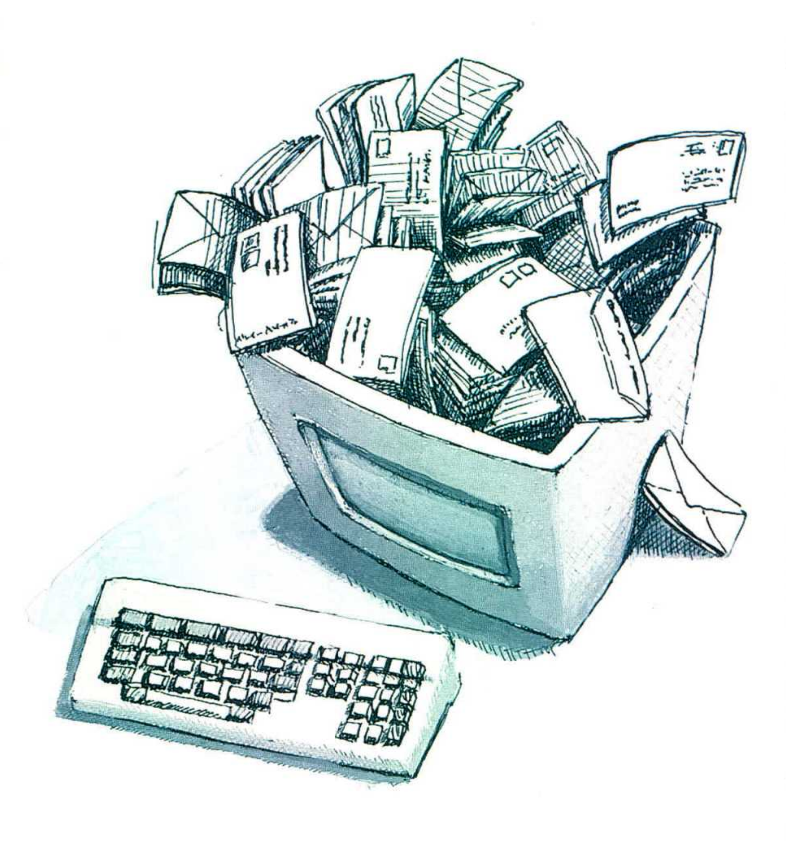
Natürlich können Sie Michael Bernards auch selbst fragen, er ist in den wichtigen Netzen erreichbar:

MICHAEL BERNARDS@BN im Mausnetz
MICHAEL_BERNARDS@MM.ZER im Zerberus-Net
MEGAMAIL:M_BERNARDS im MagicNET
2:244/11.4224 im FidoNet (Point)

oder direkt:

Michael Bernards
Bussardweg 1
5204 Lohmar-Geber

MP



Harlekin & That's Write

Vor kurzem kaufte ich mir das Multi-Accessory Harlekin, mit dem ich auch sehr zufrieden bin. Leider gibt es mit meiner Textverarbeitung That's Write Probleme beim Ausdruck seit ich Harlekin benutze. Überall auf dem Papier zeigen sich Streifen, so daß ich den Ausdruck nicht mehr verwenden kann. Starte ich meinen ST ohne Harlekin, erhalte ich auch wieder einen brauchbaren Ausdruck. Können Sie mir sagen, wie ich beide Programme zusammen zum Laufen bringe, da ich weder auf meine Textverarbeitung noch auf Harlekin verzichten möchte?

Jens Wieczorek, Essen

Red.: Die Lösung ist recht einfach. Da That's Write - wie so manche anderen Programme - den Bildschirm beim Ausdruck benutzt, gibt es Probleme mit Texten und Bildern, die direkt auf dem Bildschirm eingeblendet werden. In Ihrem Falle handelt es sich um die Uhr, die Harlekin in die rechte obere Ecke setzt. Da sie jede Sekunde weiterzählt, wird sie als unkenntliches Etwas mit dem übrigen Text zu Papier gebracht. Schalten Sie also einfach vor dem Ausdruck die Harlekin-Uhr ab und danach wieder an, und schon ist Ihr Problem gelöst.

*

Texte und Disketten

Ixch besitze seit einiger Zeit einen ATARI 1040 und will mir nun ein Textverarbeitungsprogramm zulegen. Ich habe bereits Kenntnis von Tempus 2.0. Leider ist mit diesem Programm keine Fußnotenverwaltung möglich und die eigentliche Textverarbeitung empfinde ich teilweise auch als ein wenig schwierig. Können Sie andere Programme empfehlen? Außerdem möchte ich wissen, ob Sie Programme kennen, die schreibmaschinenmäßig arbeiten, d.h. mit Direktausdruck der Zeichen oder mit mehrzeiligem Display? Wenn ja, wo kann ich so eines bekommen? Und noch eine Frage: Es gibt einige Kopierprogramme, die die Option besitzen, höher als 720 kB zu formatieren. Ich habe gehört, das sei schädlich für das Laufwerk (Dejustierung). Ich habe außerdem aus Versehen auch einmal einseitige Disketten zweiseitig formatiert: Wieso funktioniert das und kann das irgendwelche schädliche Folgen haben?

N. Willmann, Herrmannsburg

Red.: Tempus ist kein Textverarbeitungsprogramm, sondern ein Editor. Dieser kleine, aber feine Unterschied wird oft übersehen. Aus eben diesem Grund ist es auch nicht so komfortabel bei der Erstellung von Texten, dafür jedoch um so komfortabler bei der Erstellung und Bearbeitung von Quelltexten. Leider dürfen wir Ihnen kein Textverarbeitungsprogramm empfehlen. Wenn Sie jedoch durch die letzten Ausgaben der ST-Computer blättern, werden Sie einige Tests verschiedener Textverarbeitungsprogramme finden, die Ihnen Ihre Entscheidung bestimmt erleichtern.

Zu den höher formatierten Disketten: In Ausgabe 4/90 finden Sie auf Seite 84 ein Listing, das Ihnen zuverlässig die letzte benutzbare Spur Ihres Diskettenlaufwerks ausgibt. Normalerweise können keine Schäden auftreten, wenn Sie die Diskette nicht höher formatieren als es das Laufwerk verkraftet - leider ist der höchste verarbeitbare Track bei jedem Laufwerk verschieden. Mit dem Listing in der April-Ausgabe können Sie den letzten bei Ihrem Laufwerk verwendbaren Track herausfinden.

Daß Sie eine einseitige Diskette zweiseitig formatieren können, liegt daran, daß beide Diskettentypen gleich sind - mit einem Unterschied: zweiseitige sind zweiseitig geprüft, einseitige nur einseitig. Sie können also Glück haben und

Ein Wort in eigener Sache

In den Jahren, die unsere Zeitschrift existiert, haben wir immer wieder versucht, durch die Beantwortung der bei uns eingehenden Briefe ein wenig Licht in das Dunkel zu bringen, das bei der Arbeit mit dem ATARI ST schon so manch einen aus der Fassung bringen konnte - eine Tatsache, die nicht nur Ihnen, verehrter Leser, sondern auch uns oft genug zu schaffen machte. Nichtsdestotrotz haben wir uns bemüht, die Probleme zu lösen und diverse Leserbriefe zu veröffentlichen, da wir der Meinung waren, daß die jeweilige Thematik auch einen größeren Leserkreis interessieren könnte. Trotzdem gibt es immer wieder Briefe, die wir nicht beantworten können oder dürfen. Damit Sie nicht allzusehr enttäuscht zu sein brauchen oder keine Antwort erhalten, möchten wir Sie bitten, sich an folgende Spielregeln zu halten, die sich aus unserer Erfahrung ergeben haben. Fällt Ihr Brief nicht unter die folgenden Kriterien, hat er gute Chancen, positiv beantwortet oder wenigstens als Hilferuf an unsere Leserschaft gedruckt zu werden.

1. Leider gehen immer wieder Briefe mit dem Wunsch ein, ein Produkt für diesen oder jenen Anwendungsfall vorzuschlagen, verschiedene Produkte bezüglich der Vor- und Nachteile gegeneinander abzuwägen und zu bewerten. Es ist uns aus Wettbewerbsgründen nicht erlaubt, ein bestimmtes Produkt zu favorisieren, selbst wenn wir das eine oder andere in der Redaktion überzeugt einsetzen. Wir können Sie in diesem Fall ausschließlich auf die von uns möglichst objektiven Tests und eventuell anstehende Fachmessen hinweisen. Bedenken Sie bitte, daß auch wir nicht jede Textverarbeitung, jedes Malprogramm und so weiter kennen und bestimmte Produkte dadurch in das Abseits drängen würden.

2. Oft erreichen uns Briefe, die sich positiv oder auch negativ über bestimmte Händler, Softwarehäuser oder deren Produkte auslassen. Sicherlich interessieren uns solche Bemerkungen. Bitte haben Sie aber Verständnis, daß wir weder Lob noch Tadel abdrucken dürfen, da diese Aussagen meist subjektiv sind. Anders sieht die Sache beispielsweise bei Gerichtsurteilen aus, die Sie, verehrte(r) Leser(in), erfochten haben.

3. Aufgrund der Vielzahl an Briefen, die uns täglich erreichen, sind wir leider nicht in der Lage, Programmfehler anhand von Listings oder ähnlichem zu korrigieren. Dennoch sollte ein Problem möglichst detailliert beschrieben sein, denn Ferndiagnosen sind prinzipiell sehr schwer, jedoch mit genauerer Angabe der Symptome eventuell durchführbar.

4. Von Zeit zu Zeit erreichen uns Briefe mit der Bitte, die Adresse des Lesers zwecks allgemeiner Kontaktaufnahme zu veröffentlichen. Würden wir dies in die Tat umsetzen, würde sich der Umfang des anderen redaktionellen Teils beträchtlich verkleinern. Ausnahmen stellen Leser in fernen Ländern dar, für die eine Kontaktaufnahme im eigenen Land recht schwierig ist.

Zum Schluß sollen ein paar Tips eventuell voreilig geschriebene Briefe verhindern.

1. Wenn Sie ein Problem bezüglich einer bestimmten Problematik haben oder an einem bestimmten Produkt interessiert sind, finden Sie interessante Artikel darüber eventuell in vorhergehenden Ausgaben userer Zeitschrift. Zur Auswahl eignet sich das Jahresinhaltsverzeichnis besonders gut, das immer am Jahresende in der ST Computer abgedruckt wird.

2. Sollten die Probleme mit der Handhabung eines Produktes zu tun haben, wenden Sie sich zunächst an Ihren Händler und über diesen an den Distributor beziehungsweise an das Software-Haus. Die Wahrscheinlichkeit, daß Ihnen das Software-Haus weiterhelfen kann, ist um ein Vielfaches höher als die, daß wir Ihnen helfen können.

3. Lesen Sie aufmerksam die Leserbrief-Seite. Viele Fragen wiederholen sich immer wieder, obwohl wir bestimmte Probleme schon mehrfach angesprochen haben.

die Disketten lassen sich beidseitig formatieren, wenn Sie allerdings Pech haben, ist eine Seite defekt. Ein Schaden kann dadurch allerdings nicht entstehen.

*

Kompatibilität

Da ich gerne mit umfangreichen RAM-Disks arbeite, weil ich nicht glücklicher Besitzer einer Festplatte bin, spiele ich seit einiger Zeit mit dem Gedanken, mir für meinen 1040 STFM eine Speichererweiterung zuzulegen. Abgesehen von der reinen Kostenfrage (irgendwann wird die Harddisk dann ja sogar billiger) beschäftigt mich die Frage, inwieweit eine solche Speichererweiterung Nachteile bezüglich der Kompatibilität mit Programmen mit sich bringt (ist dies als Problem relevant?). Außerdem ist mir zu Ohren gekommen, daß u.U. mit erheblichen Geschwindigkeitseinbußen gerechnet werden muß. Da ich hierüber jedoch bislang z.T. widersprüchliche Aussagen erhalten habe (bis zu 25% langsamer meint der eine, Verlangsamungen wirken sich, wenn überhaupt, nur beim Bootvorgang aus, sagt der andere), hätte ich's nun gern mal genauer. Was ist dran an der Sache mit dem "lahmen Rechner"?

Ralf Arnold, Düsseldorf

Red.: Sie sind einem dicken Hund aufgesessen. Durch eine Speichererweiterung wird ein Computer niemals langsamer. Nur direkt nach dem Einschalten ist es möglich, daß der Rechner erst ein bis zwei Sekunden später den Bildschirm einschaltet - also kaum zu bemerken. Allerdings sollten Sie bedenken, daß RAM-Erweiterungen und Festplatten wie Äpfel und Birnen sind. Haben Sie tatsächlich viel zu speichern und zu laden, sollten Sie auf eine Festplatte hinarbeiten, denn sie arbeitet auch mit einem geringeren Rechnerspeicher zuverlässig, schnell und praktisch - was man von RAM-Disks nicht behaupten kann. Denn sie sind leer, sobald man (unbeabsichtigt) dem Rechner die Lebensgrundlage (Strom) wegnimmt.

*

USA-ST

Ich lebe in den USA und benutze einen deutschen ATARI ST 520+. Da es nun logisch war, einige Software hier zu kaufen, stellte ich jedoch bald fest, daß einiges nicht auf dem deutschen Modell lief (umgekehrt läuft z.B. Signum2! nicht auf einem amerikanischen Modell, welches ich an einem anderen Ort zeitweise benutze). Ich meine, mich dunkel erinnern zu können, daß es irgendwann einmal eine kleine Notiz gab, es gebe eine aufsetzbare Bank, mit deren Hilfe es möglich sei, zwei ROM-TOS-Versionen umschaltbar zu nutzen.

Jan Menge, White Plains / New York

Red.: Ihre Erinnerung trügt Sie nicht. In der Februar-Ausgabe dieses Jahres finden Sie auf Seite 179 in den Quicktips eine Anleitung, wie zwei ROM-TOS-Versionen auf EPROMs umschaltbar installiert werden können.

*

Bilderklau

Da gibt es Scanner mit 200 bis 1200 dpi und eine große Anzahl von Software, die sich besonders gut zum Kopieren von Drucksachen aller Art verwenden läßt. Hier wird sozusagen eine Anleitung zu einer Straftat geliefert, denn ich glaube, es wird fleißig gescannt, ohne auch nur einen Schimmer von Copyrights zu haben. Wenn ich ein Produkt zum Druck (in meinem Fall Speisekarten) gebe, habe ich oft viel Geduld und Zeit investiert und möchte nicht, daß eine zweite Person ohne meine Erlaubnis diese Arbeit kopiert. Neulich habe ich mir ein Buch mit Grafiken gekauft. Habe ich auch die Rechte der Kopien erstanden, oder muß ich die Grafiken eher als Leihgabe zum Anschauen betrachten? Gibt es Faustregeln oder Gesetze, die diese Vorgänge genau regeln?

Ludger Grawe, Höxter-Bosseborn

Red.: Genaue Informationen zu diesem Thema kann Ihnen nur ein Rechtsanwalt geben. Grundsätzlich sollten Sie Grafiken aber so behandeln wie Software, also nach dem Urheberrechtsschutzgesetz urteilen. Danach dürfen Sie zwar Kopien für Ihren eigenen Bedarf anfertigen, diese aber weder weitergeben noch anderen Personen zeigen. Dieser Hinweis ist jedoch eher eine Faustregel und nicht als Rechtsberatung aufzufassen.

*

Störrischer Drucker

Ich habe eine Frage meinen ATARI SMM804-Drucker betreffend. Mit ihm bin ich eigentlich sehr zufrieden, vor allem wegen der einfachen Hardcopy. Auch mit den vielen Programmen arbeitet er gut zusammen, nur ein Problem bereitet mir Kopfzerbrechen: Mit Signum2! kann ich keine Texte ausdrucken. Hier ist mein Drucker, obwohl Epson-kompatibel (oder sollte ich mich irren) recht störrisch. Er macht einen zeilenvorschub, wo eigentlich keiner sein sollte, d.h. er druckt zuerst die obere Hälfte einer Zeile, dann druckt er eine Leerzeile und bringt erst dann die untere Hälfte zu papier. Nun meine Frage: Was kann ich tun, um meinen Drucker doch noch zu einem leerzeilenfreien Ausdruck zu überreden? Ist bei meinem Problem überhaupt etwas machbar? Wenn nein, warum? Wenn ja, wie?

Gerald Schlosser, Kastl

Red.: Leider kennen wir Ihren Drucker auch nicht. Meistens ist es allerdings so, daß man an einem DIP-Schalter den automatischen Zeilenvorschub ein- und ausschalten kann. Diesen DIP-Schalter müssen Sie suchen. Existiert er nicht, gibt es bestimmt einen Befehls-Code, der den Drucker veranlaßt, den automatischen Zeilenvorschub auszuschalten. Nützen all die Einstellungen nichts, sollten Sie sich direkt an ATARI oder Application Systems wenden.

*

Wabernder Monitor

Ungefähr vier Monate nach Erwerb meines Mega ST 1 im Februar 1989 trat ein Phänomen auf, das ich am besten so beschreibe: Etwa eine halbe Stunde nach Einschalten des Computers beginnt das Bild auf dem Bildschirm zu "springen", und zwar nur in vertikaler Richtung; zuerst nur alle paar Sekunden, später aber springt das Bild mit einer Frequenz von zirka 10 Bilder/Sekunde und einer Auslenkung von ca. 1 cm in beiden Richtungen vertikal um die Mittellage, so daß es mir unmöglich ist, den Mauszeiger richtig zu positionieren oder Schrift zu lesen. Ich muß das Gerät dann ausschalten, warten und nach 2 Stunde wieder einschalten, um wieder 2 Stunde arbeiten zu können. Mein Händler riet mir, einen Lüfter einzubauen, worauf das Problem auch behoben war. Nachdem ich die Anlage jetzt in ein Tower-Gehäuse eingebaut habe, treten, trotz Lüfter, die Probleme in voller Schönheit wieder auf. Den Gang zu einem Fachmann scheue ich noch, da er wahrscheinlich mit einigen Kosten verbunden ist und ich mir im Augenblick größere Extraausgaben nicht leisten kann.

Georg-Friedrich Choitz, Hannover

Red.: Ihr Problem hört sich nach einem Hitzeproblem an. Wenn Sie den Gang zum Händler nicht wagen wollen, bleibt nur eine Möglichkeit: Kältespray. Leider kann der Fehler aber bei allen möglichen Bauteilen des Rechners liegen, so daß wir Ihnen keine Empfeh-

lung geben können, ein bestimmtes auszutauschen; Sie müssen es selbst finden. Lassen Sie den Rechner laufen und warten Sie, bis der Fehler auftritt. Dauert Ihnen das zu lange, können Sie auch einen nicht zu heißen (!) Fön benutzen. Sobald der Fehler aufgetreten ist, sprühen Sie nach und nach alle Bauteile des Rechners einzeln mit Kältespray ein. Ist der Fehler nicht mehr zu sehen, haben Sie wahrscheinlich das defekte Bauteil gefunden. Lassen Sie den Rechner wieder warm werden und sprühen nun nur das zuvor gefundene Bauteil ein. Verschwindet der Fehler wieder, haben Sie mit Sicherheit das defekte Bauteil gefunden. Tauschen Sie es gegen ein neues aus und Ihr ST wird wieder jungfräulich arbeiten. Wenn Sie sich diese Prozedur nicht zutrauen, wird Ihnen der Weg zum Händler nicht erspart bleiben. Tip: Lassen Sie sich einen Kostenvoranschlag anfertigen, dieser kostet ungefähr DM 60,- und kann später vom Reparaturpreis abgezogen werden (vorher nachfragen; wenn nicht, anderen Händler suchen!).

*

Störrischer Zeichensatz

Ich würde gerne in einem Programm bei einer zeichenweisen Ausgabe (mittels Bconout auf Device 5) die Control-Character nicht als Häkchen, ATARI-Zeichen usw. dargestellt haben, sondern als "NUL", "SOH" usw. Dazu habe ich den 8*16-System-Font hergenommen und die untersten 32 Zeichen entsprechend geändert. Es funktioniert auch das Umschalten zwischen diesem selbst erstellten und dem System-Font mittels der Funktion 'v_fontinit' (VDI ESC 102). Allerdings will ich diese Sonderzeichen auch in Dialogboxen verwenden, und da wird trotzdem der System-Font verwendet. Leider konnte ich keine Informationen finden, die dieses Problem lösen. Wie kann man hier vorgehen? Einige Herumspielereien ergaben, daß der ST offenbar gar nicht das im ROM befindliche Fontimage zur Zeichenausgabe verwendet, sondern diesen Zeichensatz in einen RAM-Bereich kopiert und von dort aus "textblittet". Ich habe meinen Font an diese Adresse geschrieben, und siehe da, er wurde wirklich vom System verwendet. Der große Nachteil: Die Adresse verschiebt sich in Abhängigkeit der geladenen Accessories. Wo finde ich die Adresse, die auch das TOS benutzt?

Wolfgang Mayerwieser, Graz

Red.: Um die Adresse zu erhalten, müssen Sie sich durch mehrere Pointer quälen. Von GFA-BASIC aus geht das so:

```
linea0$=MKI$(&HA000)
+MKI$(&H2009)+MKI$(&H4E75)
linea0%=V:linea0$
fonttab%=C:linea0%()
fhadr1%={fonttab%+4}
fhadr2%={fhadr1%+84}
```

In der Variablen fhadr2% befindet sich nach Programmende der Pointer auf den 16x8-System-Font.

*

Ereignispuffer leeren

Vor einigen Ausgaben hat ein Leser gefragt, wie er den Ereignispuffer leeren kann. Vom "Kummerkasten-Onkel" kam außer der lapidaren Antwort "Wozu soll das gut sein?" nur wenig hilfreiches. Hier erst einmal die Klärung, wozu es (bei GFA-BASIC) gut sein kann. Es gibt verschiedene Situationen, in denen Ereignisse ablaufen, von denen man nichts wissen will, z.B. die Steuerung des Fensterinhalts mit den Cursortasten. Um ein recht langes Nachlaufen des Fensterinhalts zu vermeiden, muß man alle Tastaturereignisse, die während des Bildaufbaus stattfinden, ignorieren. Das übliche Löschen des Tastaturpuffers hilft nicht mehr, wenn man die Tastatur von GEM überwachen läßt (s. Listing 1).

```
1: PROCEDURE leere_tastaturpuffer
2:    WHILE EVNT_MULTI(&X100001,0,0,0,0,0,0,0,0,
                0,0,0,0,0,adresse%,3)=1
3:    ' Warte jeweils 3ms auf Timer- und Tasta-
4:    ' turereignis, bis GEM kein Tastaturer-
5:    ' eignis mehr meldet
6:    WEND
7: RETURN
```

Listing 1: Warten auf die Tastatur

Oder man hat mehrere Fenster offen und will diese auf einen Schlag schließen, dann erfolgen u.U. mehrere Redraw-Ereignisse, die nicht mehr sinnvoll sind (Listing 2).

```
1: PROCEDURE leere_nachrichtenpuffer
2:    WHILE EVNT_MULTI(&X110000,0,0,0,0,0,0,0,0,
                0,0,0,0,0,adresse%,3)=16
3:       ' Warte jeweils 3ms auf Timer- und Nach-
4:       ' richten-Ereignisse, bis GEM kein Nach-
5:       ' richten-Ereignis mehr meldet
6:    WEND
7: RETURN
```

Listing 2: Warten auf eine Nachricht

Der Wert von 3 Millisekunden ist nur empirisch ermittelt, d.h. bei mir erfüllt er seinen Zweck. Ob er bei einem anderen Problem vergrößert oder verkleinert werden muß, muß jeder selbst prüfen.

Ralph Einfeldt, Blender

Red.: Dem ist nichts hinzuzufügen!

*

RCS?

Da wir sehr große Schwierigkeiten mit dem RCS von Digital Research, Version 2.1, haben (Chaos bei mehr als 256 Objekten pro Baum), sind wir fieberhaft auf der Suche nach einem RCS ohne diese Probleme. Würden Sie uns bitte einen Hinweis über ein fähiges RCS (möglichst als PD, schnell verfügbar) geben? Wie sieht es mit dem RCS von ATARI (Versionen 1.4 und 2.0, siehe ST-Computer 1/90) aus? Wir hätten gerne Bezugsquelle und Preis (wie gesagt: sehr schnell verfügbar, es eilt). Vielen Dank im Voraus!

iat, Gottmadingen 3

Red.: Ihr Problem liegt wahrscheinlich nicht an den 256 Objekten, sondern in der Beschränkung der Dateigröße auf 32 kByte. In dieser Ausgabe haben wir einen großen RSC-Vergleichstest veröffentlicht, aus dem Sie alle für Sie wichtigen Bezugsquellen entnehmen können.

*

ST-Programme im MAC-Stil

In letzter Zeit sieht man bei professionellen Anwendungen immer häufiger MAC-ähnliche 'Selected'-Symbole in Menüs bzw. Auswahlboxen etc. Wie etwa 'ausge-ixte' Kästchen, runde, markierte Knöpfe usw.

Wie können solche und andere - der eigenen Phantasie entsprechende - Symbole erstellt und verwandt werden auch in bezug auf selbsterstellte Resource-Dateien und deren Verwendung?

W.Vorwerk, Kiel

Red.: Solche 'eigenen' Resourcen erfordern auch eine eigene Resource-Verwaltung. In der Regel lassen die Programmierer dabei das AES, das ja beim ST u.a. für die Menüs und Dialogboxen zuständig ist, völlig links liegen und programmieren eigene Routinen zur Verwaltung ihrer Benutzerführung.

Immer up to date

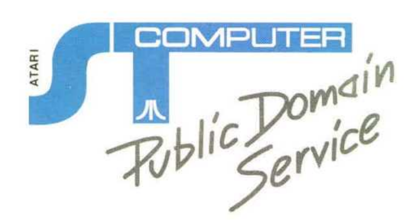
Programmname	Version	Daten
Adimens ST	3.0	N HM
Adiprog SPC Modula	1.1	N HM
Aditalk ST	3.0	N HM
Adress ST / Check ST	1.0	N H
Afusoft Morse-Tutor	2.0	N HML
Afusoft Radio-Writer	1.0	N HML
Afusoft Radiofax plus	1.0	N HML 1M
AIDA	1.1	N HM
AnsiTerm	1.4	N
Assembler Tutorial	1.06	N HM
Banktransfer	1.0	N H
1st BASIC Tool	1.1	N HML
BTX/VTX-Manager	3.0	N H 1M
Calamus	1.09.2	N H 1M
Cashflow	1.0	N H 1M
Chips At Work	1.0	N HM
CIS-L&G	1.01	
Creator	1.1	N H
Diskus	1.02	N HM
dBMAN	5.10	N HML
Easytizer	1.0	N HM
Easy Rider Assembler	2.04	N HM
Easy Rider Reassembler	2.31	N HM
fibuMAN	3.0	N H
fibuSTAT	2.3	N H
Flexdisk	1.3	N HML
FM-Meßtechnik	1.0.b	N HM
FTL Modula-2	1.18	N HM
Gadget	1.2.5b	N H
GEMinterface ST	1.1.	N HML
GFA-Artist	1.0	N L
GFA-Assembler	1.2	N HML
GFA-BASIC 68881	1.3	N HML
GFA-BASIC-Compiler	3.5	N HML
GFA-BASIC-Interpreter	3.5	N HML
GFA-Draft plus	3.0	N
GFA-Farb-Konverter	1.2	N H
GFA-Monochrom-Konverter	1.2	N ML
GFA-Objekt	1.2	N HM
GFA-Starter	1.1	N HML
GFA-Vektor	1.0	N
G+Plus	1.4	N HML
GrafStar	1.0	N H
Hanisch Modula-2	N.1	N HML
Hard Disk Accelerator	1.0	N HML
Hard Disk Sentry	1.10	
Hard Disk Toolkit	2.0	N HM
Harddisk Utility	2.2	N HM
Harlekin	1.0	N H 1M
Imagic	1.1	N HML
Intelligent Spooler	1.10	N HML
Interlink ST	1.89	N HM
Junior Prommer	2.40	N HM
K-Resource	2.0	N HM
Kleisterscheibe	2.2	N HM
Label ST	1.0	N HML
Laser C (Megamax)	2.1	N HML
1st_Lektor	1.2	N HM
Lern ST	1.22	N HML
Link_it GFA	1.1	N HML
Link_it Omikron	2.0	N HML
MagicBOX ST	7.71b	N H 1M
Mathlib	3.0	N HM
Mega Paint II	2.30	N H 1M
Mega Paint II Professional	2.31	N H 1M
Megamax Modula 2	3.5	N HM
MGE Grafikkarte	1.14	N
MGP GAL-Prommer	1.03	N H
Micro C-Shell	2.70	N HM
MT C-Shell	1.2	N HM 1M
Multidesk	1.82	N HML
Musix32	1.01	J H
NeoDesk	2.05	N HML
Omikron Assembler	1.86	N HML
Omikron BASIC-Compiler	3.06	N HML
Omikron BASIC 68881-Compiler	3.06	N HML
Omikron BASIC Interpreter	3.03	N HML
Omikron DRAW! 3.0	3.01	N HML
Omikron EasyGEM-Lib	1.0	N HML
Omikron Maskeneditor	1.0	N HML
Omikron Midi-Lib	2.1	N HML
Omikron Numerik-Lib	1.2	N HML
Omikron Statistik-Lib	1.5	N HML
PAM's TERM/4014	3.012e	N H
PAM's TurboDisk	1.7	N HML
PAM's NET	1.1	N HML
PCB-layout	1.19	N H
PC ditto Euroversion	3.96	N HML
PegaFakt	1.3	N H
phs-BTX-Box	6.0	N HML 1M
phs-ST-Box	1.2	N HM
phs-Boxtalk	1.0	N HM 1M
phs-Boxedi	1.0	N HML 1M
phs-Cheapnet	1.2	N HM
Platon	1.45	N H
Prospero Pascal	2.151	N HML
Prospero Fortran	2.152	N HML
Prospero C-Compiler	1.142	N HML
Prospero Developers Toolkit	1.103	N HML
Protos	1.1	N H 1M
Quick_Dialog	1.0	N HM
ReProk	1.10	N H 1M
Revolver	1.1	N HML 1M
Search!	2.0	N HM
Signum! zwei	1.0	N H
Skylink	1.5	N H 1M
Skyplot+	4.1	J H 1M
Soundmachine II	1.0	N HM
SoundMerlin	1.01	N HM
SPC-Modula-2	2.0	N HML
Spectre 128	1.9	J HM
1st_Speeder 2	1.0	N HML 1M
SPS ST	1.5	N H 1M
STAD	1.3+	N H
Steuer-Tax 2.9	1.10	N HM
Steuer-Tax 3.9	1.10	N HM
STop	1.1	N HM
ST Pascal plus	2.08	N HM
SuperScore	1.4	J H 1M
Tempus Editor	2.05	N HM
Theca Librarian	1.0	N HM
TIM	1.2	N H
TIM II	1.0	N H 1M
Transfile ST 1600	1.1	N HM
Transfile ST 850	1.1	N HM
Transfile ST plus	3.0	N HM
Turbo C	1.1	N HM
Turbo ST	1.8	N HML
UIS II + Hermes	2.5	
V_Manager	3.02	N H
VSH Manager	1.0	N HML 1M
WERCS Resource-Editor	1.0	N HM

Irrtum vorbehalten!

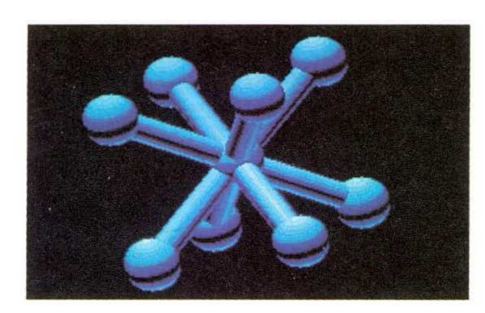
Daten-Legende : N = kein Kopierschutz, J = Kopierschutz, H = hohe Auflösung, M = mittlere Auflösung, L = niedrige Auflösung, 1M = mindestens 1 Megabyte

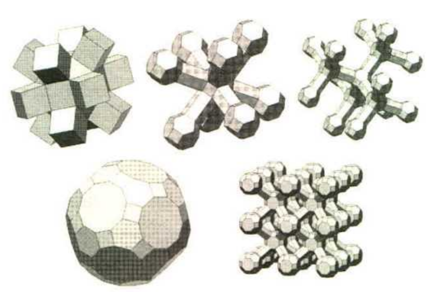
Inserentenverzeichnis

ST-COMPUTER PUBLIC DOMAIN

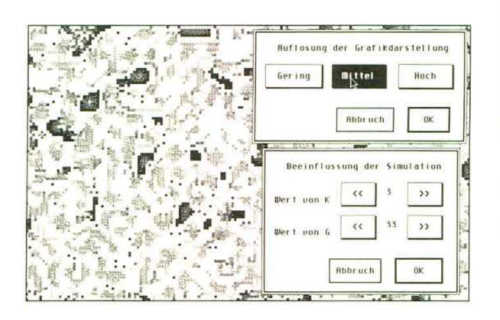


PD 325





REGULÄRE FIGUREN: Mehrere verschiedene Programme, die zum Beispiel platonische, archimedische und rhombische Körper in der Axonometrie in beliebiger Lage darstellen, sowie reguläre, räumliche Strukturen. Das klingt recht kompliziert, das Ergebnis ist allerdings sehenswert. Mit Animationsteil.



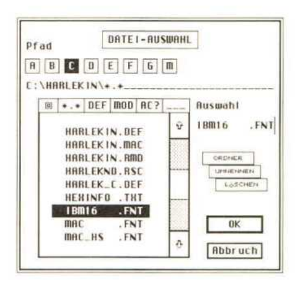
ZELLEN: Zellen ist ein Programm zur Simulation von Abläufen, ähnlich wie bei chemischen und biologischen (z.B. Life) Reaktionen. Man geht von einem beliebig großen, zweidimensionalen Feld aus, dessen Elemente verschiedene Zustände erreichen können, die nach einem bestimmten Schema berechnet werden.



326 UTILITIES

AZ_D: Autozeit dient dazu, die Systemzeit laufend fortzuführen, auch wenn der Rechner nicht über eine permanente Uhr verfügt.

HELFER: Vielfältiges Accessory zur Organisation des Rechners. Dieses Accessory besteht aus sehr vielen nützlichen Tools, z.B Bildkonvertierung, Snapshot und jeder Menge GFA-Tools.



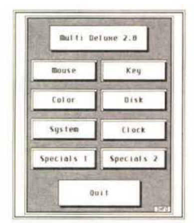
EX_FILESELECT: Fileselectorbox mit vielen interessanten Features. Hängt sich ins System und ersetzt die normale Box.

- Laufwerksauswahlknöpfe für bis zu 13 Laufwerke
- fünf definierbare Datei-Extensions-Wahlknöpfe
- Anlegen von Ordnern, Umbenennen und Löschen von Dateien
- Disk-Information (freier Platz in Bytes)
- Laufwerk-Ordner-Pfad bis zu 80 Zeichen Länge
- erweiterte Datei-'Joker' wie z.B. "*ABC??*.*S*"
- Direktaufruf über den Accessory-Eintrag



UHR: Nette Uhr, die sowohl digital, als auch alalog erscheint. Die Anzeige wird automatisch (auch nach Verlassen eines Programms) in den Fordergrund gebracht.

MINI_ORC: Accessory zur Schrifterkennung. Durch Einrahmen eines bestimmten Bildschirmausschnitts (Grafik bzw. laufendes Programm) wird der dort enthaltene Text als ASCII-Datei erfaßt bzw. gedruckt.



Multi Delux: Accessory mit mehreren nützlichen Features. Z.B. Maus-Speeder, Zeitanzeige, Systeminfos, Stepratenumschaltung, Tastenreset, Mauspositionsanzeige...



SCREENSAVER: Bildschirmschoner und Maus-Speeder

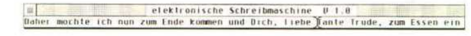
DESKTIP: installiert Desktop.inf nachträglich oder schreibt das momentan aktive Desktop.inf aus dem Speicher auf Disk.



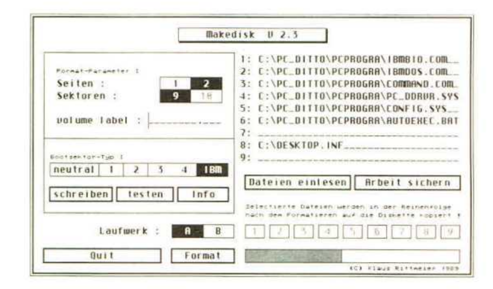
327 ACCESSORIES

Erste und zugleich geballte PD-Einsendung aus der DDR. Jede Menge toller Accessories, die man gut gebrauchen kann.
ARCSHELL, BIORHYTM, BOOTMENU, CHK_TREE, EXE, FRACTAL, MAKEDISK, PRN_SEND, TERMIN, TINY_EDI, TYPEWRIT, X_FORMAT, XDIR, X_INFO.
z.B:

TINY_EDIT: Editor als Accessory. Jederzeit verfügbarer Editor, der in eigenem Fenster auf dem Desktop oder in beliebigen Programmen arbeitet.



TYPEWRIT: Einzeilige Schreibmaschine. Der eingegebene Text wird nach jeder Zeile zum Drucker geschickt. Korrektur möglich. Sehr praktisch für kurze Briefe oder zum Ausfüllen von Formularen.

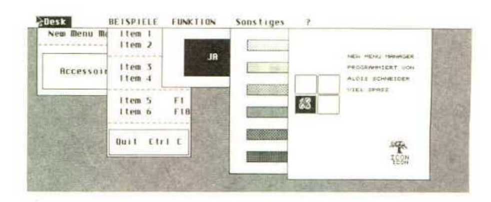


MAKEDISK: formatiert Disketten (auch IBM) und kopiert sodann automatisch bestimmte Programme und Dateien darauf (z.B. IBM-System-Files oder Autoordner).



LASER.LIB: mehrere Libraries für LASER C. Die Libraries enthalten unzählige nützliche Funktionen, die zur Unterstützung der Original-Libraries benutzt werden können.

- Funktionen zum Verarbeiten von Zeichenketten
- Adreßbereiche vergleichen und kopieren
- Funktionen zum Steuern der Bildschirmausgabe
- Terminalroutinen
- Grafik
- Funktionen zum Erstellen und Bearbeiten von Menüs
- Popup-Menü aufbauen und ausführen
- Funktionen zum Bearbeiten von Datenstrukturen
- Stack-Datenstruktur aufbauen und manipulieren
- Programmaufrufargumente prüfen und Variablen zuordnen
- Systemkommando ausführen



NEW MENU MANAGER: Routinensammlung zur Programmierung einer alternativen Menüsteuerung. Der Unterschied: Beim Verlassen der Drop-Down-Menüs muß kein Mausklick mehr erfolgen, die Menüs schließen automatisch. Natürlich können auch noch einige andere Raffinessen programmiert werden. Die Menüs lassen sich weiterhin mit dem RCS erstellen. Kompletter Source in C.

CMPLX_16.FOR: ist eine Programmsammlung für doppelt-genaue komplexe Ausführung der INTRINSIC-Funktionen von Fortran. Darüber hinaus enthält sie viele in Fortran 77 nicht vorgesehene Funktionen.

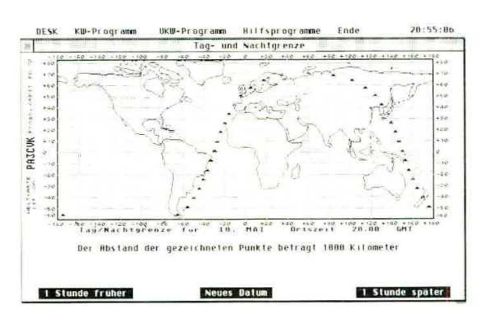


Morsezeichen entschlüsseln zu können, ist keine Kunst - reine Übung, doch ein Computer ist ideal für diese Aufgabe.

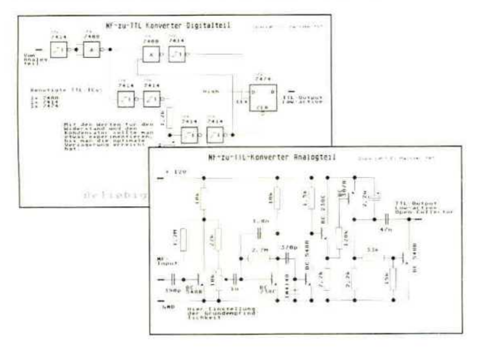
MORSEDEKODER: Programm zum Übersetzen von Morsezeichen. Über einen Soundsampler (wird benötigt) werden die akustischen Signale in den Rechner eingespielt und von diesem entschlüsselt.

MORSEMASTER: Trainingsprogramm für Morsezeichen

PEILEN: Amateurfunkprogramm. Bestimmung von Peilschnittpunkten in leicht erstellbaren Hilfskarten zur räumlichen Orientierung. Diese einmal erstellten Karten können in beliebigem Maßstab dargestellt werden.



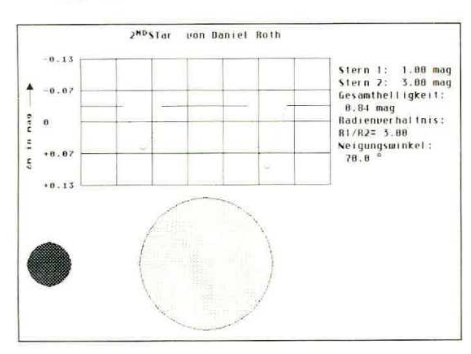
Anzeige von Städten (200) und Relaisfunkstellen (100), welche schon in 2 Dateien beiliegen. Entfernungsberechnung, Locatorbestimmung und Antennenrichtung der anklickbaren Punkte. Abspeichern von mehreren Eigenstandorten. Für Kurzwellenverbindungen: Suchen nach Rufzeichen oder Ländern auf einer Weltkarte. Berechnung von Antennenwinkel, Entfernung und Markierung des Großkreises. Einzeichnung der Tag- und Nachtgrenze. (s/w)



MDEC: umfangreiches Programm zur Erkennung von Amateur- und Wetterfunk. Der komplette Schaltplan der Hardware ist auf Diskette enthalten (Analog- und Digitalteil). Damit ist es möglich, die Signale einem normalen Radio zu entnehmen, und dabei die anfallenden Störsignale zu filtern. Natürlich kann das Programm auch als Morsetrainer benutzt werden, wobei der Rechner 'eingeklickten' Text entziffert.



330 ASTRO



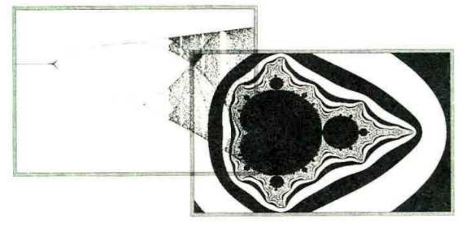
2ndSTar: Berechnung der Lichtkurve eines aus zwei Sternen bestehenden Systems, deren Komponenten sich gegenseitig bedecken und so einen Lichtwechsel hervorrufen.

AstroPho 1.0: Verschiedene Berechnungen zur Photographie astronomischer Objekte

Kalender 1.42: Berechnung von Tagen zwischen Daten, Feiertagen, Halbjahreskalender auf Bildschirm, Jahreskalender auf Drucker, Feiertage, Mondphasen, u.v.m.

Sonnenuhr 1.2: Berechnung von Zifferblättern ebener Sonnenuhren. Grafische Darstellung auf Bildschirm und Drucker. Letzteres in Signum!-Qualität.

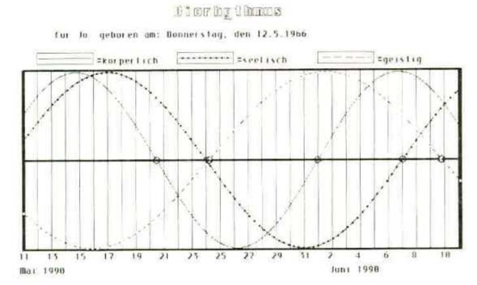
StarTrek: Ein vielleicht unspielbares Spiel. (s/w)



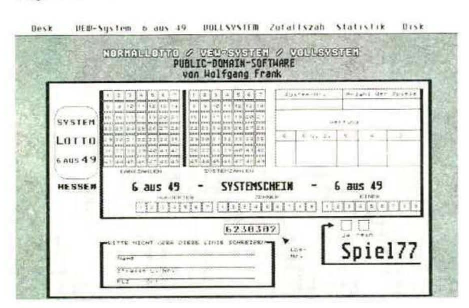
Chaos: Grafische Darstellung von Feigenbaum- und Mandelbrot-Mengen.

Starfind: Sehr gut gestaltetes Programm zur Orientierung am Sternenhimmel. Man geht mit der Maus auf einen Stern oder Himmelskörper, und das Programm teilt Ihnen nun mit, wie der Himmelskörper heißt. (s/w)

331



Hier sind wieder zwei Biorhythmus-Programme, die sehr gut gestaltet sind. Man kann verschiedene Rhythmen vergleichen sowie alle Eigenschaften ausdrucken lassen. Die Auswertung der Kurven erfolgt nach bestimmten Kriterien, wie z.B. körperlich, seelisch, geistig, oder nach deren Mittelwert.



LOTTO: Das andere Lotto-Programm, denn mit diesem Programm können Sie sich auch mit dem Systemtippen befassen. Es ist eine nützliche Hilfe bei der Auswertung aller VEW-Systeme. (s/w)

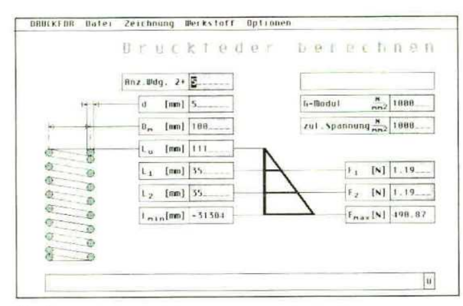
LOTTOKAT: Ein weiteres Lotto-Programm. Es ist in der Lage, alle bisherigen Ziehungen des Samstags- und Mittwochslottos zu verwalten. Die verschiedenen Statistiken werden sehr gut als Diagramme dargestellt. Als weiteren Zusatz hat dieses Programm eine Abwägung zwischen vermutlicher Gewinnchance und dem Einsatz.

332

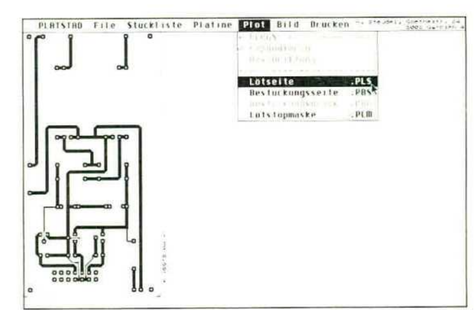
Diese Programme sind eine sehr nützliche Hilfe für den technisch versierten ST-User.

Feder: Mit diesem Programm kann man Zug- und Druckberechnungen für Federn durchführen, wobei hier zwischen 3 verschiedenen Materialien gewählt werden kann. (s/w)

--> Bild siehe nächste Spalte



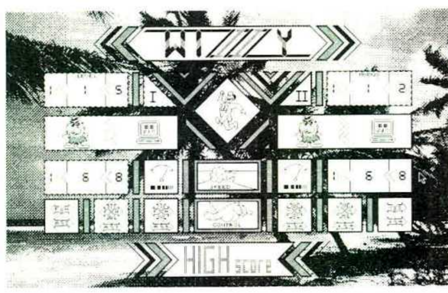
Feder



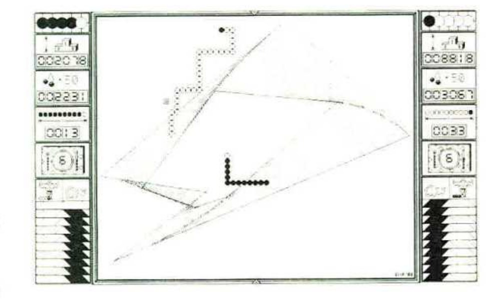
PLATSTAD ist der Draht zwischen den Programmen PLATINE ST und STAD, daher der Name. Es stellt im wesentlichen einen "Bildschirm"treiber (ähnlich Druckertreiber) dar, der Platine ST-Dateien (.PLT) auf den Bildschirm zeichnet. Die Bilder (Platinen) können dann vielseitig weiterverarbeitet werden. Dazu eignet sich jedes pixelorientierte Zeichenprogramm. (s/w)



333



Wizzy ist ein Wurmspiel, bei dem es darum geht, Nahrung aufzunehmen ohne mit dem Gegner, der Wand oder mit dem eigenen Wurmkörper zu kollidieren. Man kann wählen zwischen der klassischen Variante, d.h. Wurm stößt sich Kopf - das Spiel ist zu Ende, und dem zweiten Modus, daß der Wurm kürzer wird, wenn er sich den Kopf anstößt. Die grafische Gestaltung ist sehr aufwendig und auch die Optionen können sich sehen lassen. Die Grafiken sind sehr interessant gemacht, es gibt verschiedene Schwierigkeitgrade, verschiedene Hindernisse (z.B. bewegt) sowie Geschwindigkeitsstufen. Ein sehr schönes Programm mit hoher Spielmotivation(s/w)



ST-COMPUTER PUBLIC DOMAIN

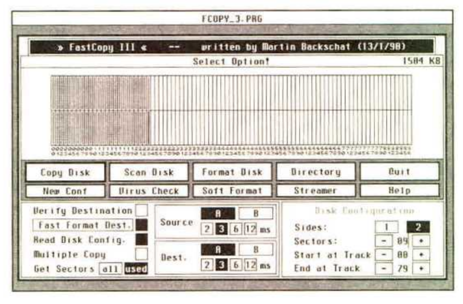
PD 306 DRUCKER



ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ
ABCDEFGHIJ

GUTENBERG: Exklusives Druckprogramm für 9-Nadeldrucker. Gutenberg bringt ASCII-, 1st_Word und Textomat-Texte mit beliebigen eigenen Zeichensätzen zu Papier, samt allen Attributen (fett, schmal...) und Proportionalschrift. Mit komplettem Source-Code (Pascal). (s/w)

PD 300 JUBILÄUMSDISK



FCOPY III: eine komplette Neuentwicklung des legendären FCOPY II. Hinzugekommen sind unzählige Features, die wir kurz anreißen wollen.

- Kopieren nur belegter Sektoren
- Formatänderung beim Kopieren
- Reparatur defekter Disketten !!!
- Formatieren (auch Hyperformat, 11 Sektoren, >900 kByte)
- Fastformat (32 sec, doppelseitig)
- Softformat (Löschen der FAT und des Directories)
- Festplatten-Backup, komplette Partitionen werden auf Disk-. kette gerettet und können später auf Platte restored werden.
- läuft auch als Accessory

UPDATES



PD 237 SAGROTAN 4.17

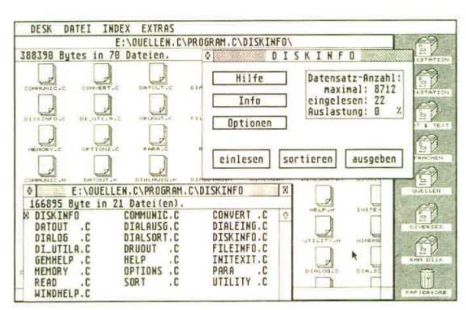
Das bekannte und sehr weit verbreitete Sagrotan ist um verschiedene Virenerkennungen erweitert worden. Leider werden ja nach wie vor Viren auf dem ST im Umlauf gesetzt. Sorgen Sie dafür, daß auch Sie up to date sind.

MACHEN SIE MIT!

Möchten Sie ein selbstgeschriebenes Programm in unsere PD-Sammlung geben, um es auch anderen Usern zugänglich zu machen? Kein Problem. Schicken Sie es uns auf einer Diskette zu, samt einer Bestätigung, daß es von Ihnen geschrieben wurde und frei von Rechten Dritter ist. Bei Fragen steht Ihnen die Redaktion gerne zur Verfügung.

MAXON Computer • ST-Computer PD
Industriestr. 26 • D-6236 Eschborn

PD 307 UTILITY



DISKINFO 3: Programm zum Einlesen und Verwalten von Disketteninhalten. Inhaltsverzeichnisse von Disketten, Partitionen oder Ordnern können eingelesen werden. Export in Adimens.

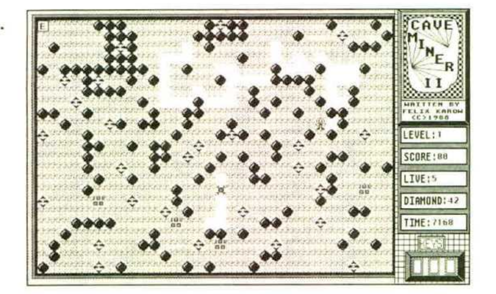
PD EN MASSE!

Wegen der großen Nachfrage haben wir sie ins Leben gerufen - die PD-News. Sie enthält die komplette Übersicht aller ST Computer-PD-Disketten. Sie erhalten sie - wie gewohnt - kostenlos gegen Zusendung eines mit DM 2.40 frankierten und an Sie adressierten *DIN A4*-Rückumschlages, denn in kleinere paßt unsere Zeitschrift beim besten Willen nicht rein!!!

MAXON-Computer • PD-NEWS
Industriestr. 26 • D-6236 Eschborn



PD 308 SPIELE



CAVEMINER II: Sehr schönes Spiel, das über mehrere Levels geht. Hier haben Sie die Aufgabe, alle Diamanten in möglichst kurzer Zeit einzusammeln, um in den nächsten Level zu gelangen, wobei die Gefahr besteht, von einem Felsen erschlagen zu werden. (s/w)

VAULT III: Klassisches Text-Adventure.

ABKÜRZUNGEN

1MB = mind. 1MB Speicher notwendig
s/w = nur Monochrom; f = nur Farbe

ATARI ST COMPUTER Public Domain Service

DIREKT-VERSAND

Alle PD-Disketten unserer Sammlung gibt es direkt bei MAXON-Computer.

Um einen schnellen Versand zu ermöglichen, beachten Sie bitte folgende Punkte:

1. Schriftliche Bestellung

- Der Unkostenbeitrag für eine Diskette beträgt DM 10,-
- Hinzu kommen Versandkosten von DM 5,- (Ausland DM 10.-)
- Bezahlung per Scheck oder Nachnahme
- (Im Ausland nur Vorrauskasse möglich)
- Ab 5 Disketten entfallen die Versandkosten (DM 5.- bzw. DM 10.-)
- Bei Nachnahme zuzüglich DM 4,00 Nachnahmegebühr

2. Telefonische Bestellung

MAXON-Computer GmbH
'PD-Versand'
Tel.: 0 61 96 / 48 18 11
Fax: 0 61 96 / 4 18 85

Mo-Fr 9⁰⁰ - 13⁰⁰ und 14⁰⁰ -17⁰⁰ Uhr

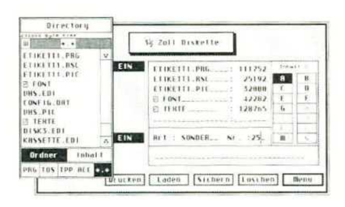
- Lieferung erfolgt per Nachnahme

Adresse:

MAXON-Computer GmbH
'PD ST-Computer'
Postfach 5969

Nutzen Sie die PD-Karte in diesem Heft





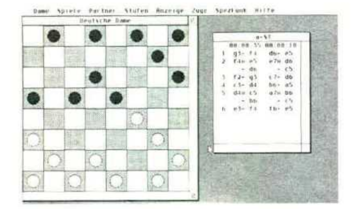
MASTER ETIKETT

Leistungsstarkes Programm zum Bedrukken von Disketten, Video- und Audio-Cassetten, sowie Paket- und Ordneraufklebern. Das Programm bietet spezielle Eingabemasken, die genau auf diese Aufkleber passen.



Bei Diskettenaufklebern können die Daten sogar bequem von der jeweiligen Diskette gelesen und per Klick in den Aufkleber übertragen werden. Mehrere Fonts und Grafikeinbindung unterstützen eine originelle Gestaltung. Druckeranpassung vorhanden.

MASTER ETIKETT
SD 25 DM 15.-



DAME

Computerumsetzung des alten Brettspiels. Bei fehlendem Gegner springt der ST gerne ein und bietet einen vollwertigen Gegner, den man erst einmal schlagen muß. Die Figuren werden per Maus angewählt, die Züge protokolliert (klassisch/kanonisch) und analysiert. Der Rechner spielt in verschiedenen Stufen, von der Blitzpartie bis zu unendlicher Bedenkzeit, mit oder ohne Handicap. Es können Zugvorschläge ausgegeben und Züge zurückgenommen werden. Auch läßt sich eine bestimmte Partie aufbauen oder die momentane abspeichern. DAME spielt nach 3 verschiedenen Varianten: Draught/Checkers, Deutsche und Polnische Dame.

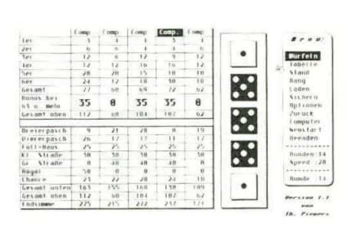
DAME
SD 26 DM 15.-

DER MOTOR

Autotechnik

Der Motor erklärt mit zahlreichen Grafiken die Funktionsweise eines Verbrennungsmotors. Sehr anschaulich sind die bewegten Grafiken, die z.B. die Bewegung eines Kolbens und die Zündzeitpunkte deutlich machen. Das gezeigte Wissen wird zusätzlich in einem Quiz abgefragt - der Lernerfolg ist dadurch fast garantiert.

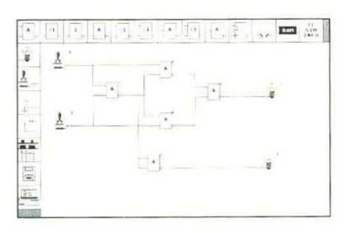
DER MOTOR
SD 20 DM 15.-



WÜRFELPOKER

Da sind die Würfelzocker gefragt. Würfelpoker kann man mit mehreren Personen (auch Computer) spielen. Sinn ist es, seine Würfe möglichst geschickt zu verteilen uns gezielt nachzuwürfeln. Neben vielen Pärchen gilt es Straßen oder sonstige Kombinationen zu erzielen.

WÜRFELPOKER
SD 27 DM 15.-



ICSIM

Logik-Simulator

Das Programm simuliert das Verhalten von logischen Schaltungen. Bausteine und Verbindungen werden frei per Maus positioniert bzw. verbunden. Eine Schaltung läßt sich somit leicht austüfteln, testen und erst dann in die Praxis umsetzen.

Es sind die Logikbausteine nach DIN 40900 enthalten: AND, OR, NOT, NAND, NOR, XOR, RS-FF, KLEMME, LAMPE, SCHALTER, 0V und +5V.

Die Simulation wird als Impulsdiagramm oder Logiktabelle ausgegeben. Weiterhin liefert das Programm den Schaltplan und eine Liste der benötigten Bauteile.

ICSIM (1)
SD 25 DM 20.-

HAUSKASSE

Kostenplanung

HAUSKASSE verwaltet die Ausgaben eines privaten Haushaltes. Man bekommt dadurch einen Überblick über die angefallenen Kosten und die verfügbaren Mittel, bzw. an welcher Stelle man sparen sollte.

Am Monatsanfang wird eine Vorausplanung vorgenommen und mit den tatsächlichen Ausgaben verglichen. Die Abweichungen werden grafisch dargestellt. Weiterhin werden fixe zu einem bestimmten Datum anfallenden Kosten, z.B. Rundfunkgebühren, verwaltet und automatisch in die Planung und zu den Eingaben übernommen. Ein ideales Hilfsmittel, um die privaten Kosten in Griff zu bekommen.

HAUSKASSE
SD 26 DM 15.-



ASSOZIATIX

Assoziative Datenbank

Assoziatix ist eine assoziativ-Muster orientierte Datenverwaltung, die es ermöglicht aus einer großen Datenmenge bestimmte Gruppen auszufiltern und daraus Zusammenhänge zu finden (z.B. Rasterfandung).

Fragebogenerhebung, -verwaltung und -auswertung: Erstellen eines Erhebungsblattes(Protokoll) am Bildschirm, Ausdruck des Formblattes, Auswertung und schnelle assoziative Suche nach bestimmten Konstellationen......

Mit Hilfe des Formulareditors können die Eingabemasken leicht am Bildschirm gestaltet und die Größe bzw. die Art der Eingabefelder bestimmt werden. Sogar Grafikeinbindung in die Masken ist vorhanden.

Einige Besonderheiten:
- Passwortschutz der Dateien (abschaltbar)
- Export- und Importfunktion von Daten, Serienbriefe
- Reportdokumentation (dokumentiert die Selektionsvorgänge und deren Ergebnisse)
- assoziative Suche und Selektion nach verschiedenen Modi
- Statistische Berechnung der numerischen Werte eines Feldes
- Expertfunktion: das Programm fragt gezielt nach einzelnen Punkten. Durch einfache Beantwortung dieser Anfragen wird versucht, eine zutreffende Lösung zu finden.
- Volltextsuche in den Textdateien möglich
- Grafikeditor: Spiegeln, Drehen, Zoomen, Balken- Linien und Kuchengrafik.

Mögliche Anwendungen wäre z.B. die Archivierung von Film, Foto- und Tonmaterial nach Merkmalen

ASSOTIATIX (2 Disketten)
SD 27 a/b DM 30.-

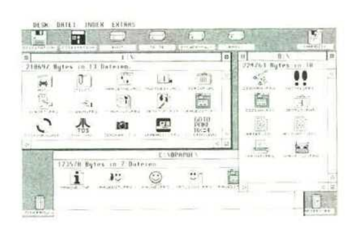


Special Paint 2

Grafik deluxe

Grafikprogramm der Extraklasse. Neben den vielen nützlichen Funktionen zeichnet sich Special Paint vor allem durch seine Geschwindigkeit, seine bequeme Bedienung und seine Kompatibilität zu bekannten Malprogrammen (z.B. STAD) aus. Special Paint bietet umfangreiche Blockfunktionen, Lasso, superschnelle Lupe, Maskierungen, Clippen, schnelle Bieg-, Zerr- und Drehoptionen, Animation und vieles mehr. Special Paint arbeitet mit dem neuen Clipboard (Übertragung zu anderen Programmen), schreibt z.B. IMG-Format für Wordplus, konvertiert alle Auflösungen und beherrscht umfangreiche Textfunktionen (ladbare Fonts, Blocksatz, Zeilenumbruch).

Special Paint
SD 21 DM 20.-



OPAQUE

Das Desktop mit neuem Gesicht

Wie wäre es mit einem zweckmäßigen und originellen Desktop? Opaque bietet die Möglichkeit, jedem Programm ein eigenes Icon zuzuordnen. Vorbei ist die Zeit, als es nur 3 Icons gab. Jedes Programm präsentiert sich nun durch ein sinnbezogenes Icon. Auch die Laufwerke lassen sich ändern. Weiterhin kann man die Icons mit Wildcards definieren, z.B. *.BAS, *.TXT, *.ACC oder gar WORD????.PRG für verschiedene Programmversionen. Im Programm sind über hundert neue Icons enthalten, eigene sind mit dem beigefügten Editor leicht zu erstellen und beliebigen Programmen/Dateien zuzuordnen.

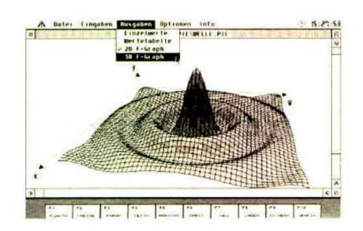
OPAQUE
SD 22 DM 15.-

Magic Box ST

Mailbox

Die Mailbox Magic Box ST ist befehlsgesteuert und bietet dem Benutzer über 80 Befehle zur komfortablen Steuerung des Systems. Magic Box dient zum Aufbau einer lokalen Mailbox, jedoch ohne Vernetzung. Das überaus komfortable Mailbox-Programm läuft auf S/W-Monitoren und allen ATARI ST mit mindestens 1 MB. Eine Festplatte wird zum Betrieb empfohlen. Alle benötigten Programme wie Pakker, User- und Brett-Editor werden dem Programm mitgeliefert.

Magic Box ST
SD 12 DM 15.-



FORMULA

2D-/ 3D-Plotter

2D-/3D-Plotter für mathematisch-wissenschaftliche Anwendung. Der eingebaute Formel-Interpreter beherrscht neben allen gängigen Operationen auch die Definition verschiedener Formeln in bestimmten Teilbereichen, logische Operationen und IF..THEN..ELSE-Kommandos. Somit kann man auch komplizierteste Terme darstellen. 3D-Grafiken lassen sich aus verschiedenen Blickrichtungen anzeigen und mit Schattierungen versehen, was den räumlichen Eindruck verstärkt.

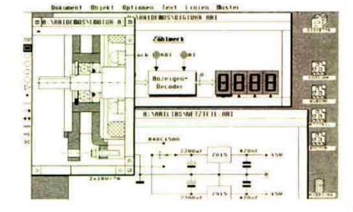
FORMULA
SD 23 DM 20.-

VIRUS EX

Virenschutz

Leider ist das Thema Viren auf dem ST ein noch aktuelles Thema, denn Computerviren sind im Umlauf und treiben ihr Unwesen. Hier gilt wie bei der Zahnpflege 'Vorbeugen'. VIRUS EX beugt doppelt vor. Zum einen überprüft es ständig die eingelegten Disketten (Bootvirus), die Systemadressen, die VBL-Queue und durchsucht den Speicher nach residenten Programmen, so daß ein sich evtl. gerade aktivierender Virus nicht unerkannt bleibt.
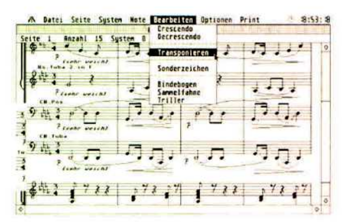
VIRUS EX
SD 07 DM 15.-



ARIADNE

ARIADNE ist ein besonderes Zeichenprogramm. Es ist objektorientiert und bietet die Möglichkeit, jedes beliebige Grafikobjekt (mit Doppelklick) zu öffnen, worauf eine neue Zeichenebene bereitgestellt wird. Die Objekte auf dieser Ebene können dann wiederum geöffnet werden usf. Diese hierarchische Struktur eignet sich besonders zur Darstellung komplizierterer Dinge, z.B. Blockschaltbilder, Schaltungen, Software-Dokumentation.

ARIADNE [1]
SD 8 DM 15,-



TRISTAN

Notensatzsystem

Für alle Musikfreunde, die nicht nur vom Blatt spielen, sondern auch aufs Blatt schreiben, bietet das Notensatzsystem TRISTAN die ideale Möglichkeit, ihre Noten professionell zu Papier zu bringen. Es lassen sich Partituren mit bis zu 100 Seiten mit max. 32 Notensystemen je Seite bearbeiten. Alle im klassischen Notensatz gebräuchlichen Zeichen, u.a. auch Schlagzeug- und Vorschlagnoten, lassen sich bequem mit der Maus edieren. Ebenfalls stehen mehrere Notenschlüssel, Sammelfahnen, Triller und Bindebögen zur Verfügung. Automatische Transponierfunktion. Ein Ausdruck ist auf 9- und 24-Nadeldruckern möglich, wobei das Druckbild durch eine interne höhere Auflösung der Zeichen (ähnlich Signum!) optimiert wird.

TRISTAN
SD 24 DM 25.-

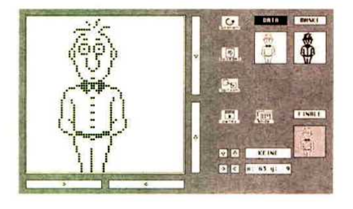
PANDA

Der Farbemulator

Farbemulator, der es erlaubt, die Farbauflösung auf einen monochromen Monitor (SM 124,..) zu simulieren. Dadurch kann man ein Farbspiel laufen lassen oder beim Programmieren austesten, ob ein Programm auch auf Farbe läuft. Emulation der mittleren und niedrigen Auslösung, Graustufenumrechnung, Abspeichern des Bildes in 32K und DEGAS-Format, variable Einstellung der Bildwiederholfrequenz. Trotz der enormen Rechenarbeit, die PANDA im Hintergrund vollbringen muß, erreicht er Geschwindigkeiten bis zu 90% je nach Bildfrequenz.

PANDA
SD 18 DM 15,-

[1] nur für monochromen Monitor (SM 124)
[2] nur für Farbmonitor



ICONDESIGN

Dieser Icon-Editor ermöglicht das komfortable Erstellen von Icons beliebiger Größe. Grafische Grundfunktionen (Linien, Kreis,..) im Icon-Raster, automatische Maskierung und Erzeugung von Icon-Daten und Source-Code zum Einbinden in die eigenen Programme machen Icondesign zu einem Tool der Sonderklasse. Auch Icon-Ausgabe als RSC-Datei. Ab TOS 1.2.

ICONDESIGN
SD 17 DM 15,-

ROBOTWAR

Robotwar basiert auf dem bekannten Corwars. Im Gegensatz dazu werden die Kampfprogramme grafisch dargestellt, Angriffe animiert und Zerstörungen bildlich gezeigt. Die Kampfprogramme werden in einer assemblerähnlichen, zudem sehr luxuriösen (Schleifen, Prozeduren...) Sprache verfaßt.

Die max. 16 gleichzeitig kämpfenden Roboter können sehen, sich vergraben, Ziele anvisieren, Bomben legen und andere angreifen.

ROBOTWAR [1]
SD 13 DM 15,-

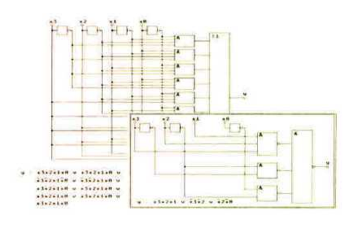
HARDCOPY II

Die erste Farb-Hardcopy für den ST

Was nutzt der beste Farbdrucker, wenn er ein Bild nur in s/w drucken kann? Was nutzt der neue 24-Nadeldrucker, wenn die ST-Hardcopy nicht funktioniert? Was nutzt ein 9-Nadeldrucker, wenn die Grafik verzerrt kommt? Was nutzt ein 18- oder 48-Nadeldrucker, wenn kein Programm ihn unterstützt?

All diese Probleme löst HARDCOPY II, denn dieses Accessory unterstützt durch seine flexible Anpassungsmöglichkeiten alle Drucker, die der ST jemals gesehen hat oder noch sehen wird.

HARDCOPY II
SD 15 DM 15.-



Quinemac

Der Schaltfunktionen-Analyzer

Um eine Schaltung zu entwerfen und zu optimieren, gibt es verschiedene und sehr aufwendige Verfahren. Bis alles so ist, wie es sein sollte, muß man sich durch etliche Gesetze kämpfen. Aber wofür das alles, wenn es mit Quinemac einfacher geht? Sie geben die Anzahl Ihrer Eingänge und den Zustand der Ausgänge an - und Quinemac liefert die dazugehörige Schaltfunktion, einen Schaltplan und dessen Kostenfaktor. Außerdem bietet er die Möglichkeit, sich eine Karnaugh-Tafel erstellen zu lassen!

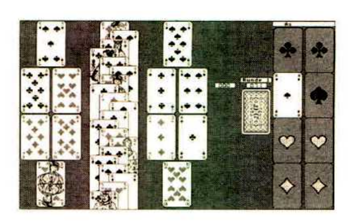
Quinemac [1]
SD 10 DM 15,-

MAKI

Das Rechenäffchen

Mathematik ist ein bei Kindern meist unbeliebtes Thema. MAKI - das Rechenäffchen - schafft da Abhilfe. MAKI wurde geschrieben, weil Lernen Spaß machen soll. MAKI ist ein Rechentrainer, der auf die Wünsche und Bedürfnisse von Kindern abgestimmt wurde. Mit MAKI lernt man die vier Grundrechenarten. Doch MAKI ist durch die einstellbare Schwierigkeitsstufe auch für Erwachsene geeignet.

MAKI [1]
SD 19 DM 15,-

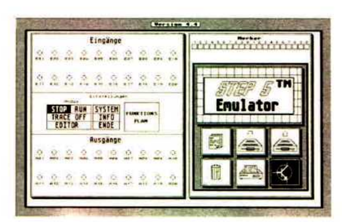


PATIENCE

Kartenspiel, das hohe Aufmerksamkeit erfordert. Es schult das Denkvermögen, fördert die Kombinationsfähigkeit, entspannt und beruhigt zugleich. Im Programm sind folgende Patience-Varianten enthalten: Standard, Eiffelturm, Zopf, Kleine Harfe, Matriarchat und Bildergalerie.

Das Programm gibt auf Wunsch Lösungsvorschläge. Eine sehr gute Grafik und ein durchdachtes Konzept lassen jedem Kartenfreund das Herz höher schlagen.

PATIENCE [1]
SD 11 DM 15,-



SPS-Emulator

für programmierbare Steuerungen

Unser SPS-Emulator baut auf einem SIEMENS PG 605-Programmiergerät in STEP 5 auf. Mit ihm lassen sich SPS-Programme schreiben, auf Simulationsbasis austesten, laden, speichern, ändern, ausdrucken und als FUP (Funktionsplan mit logischen Gattern) ausgeben. Enthalten sind ein Editor, ein Interpreter und FUP-Generator.

SPS-Emulator [1]
SD 14 DM 15,-

EASY ADRESS 3.0

Adreßverwaltungsprogramm mit Grafikeinbindung. Die verwalteten Personen auf dem Monitor gezeigt werden. Das Programm schneidet ein 'Passbild' heraus und fügt es in den Datensatz ein. Doch auch ohne Bilder kann das Programm benutzt werden. Es verfügt über alle nötigen Funktionen, z.B. Indexverwaltung, Suchen von Datenbruchstücken, Telefonlistendruck, etc.

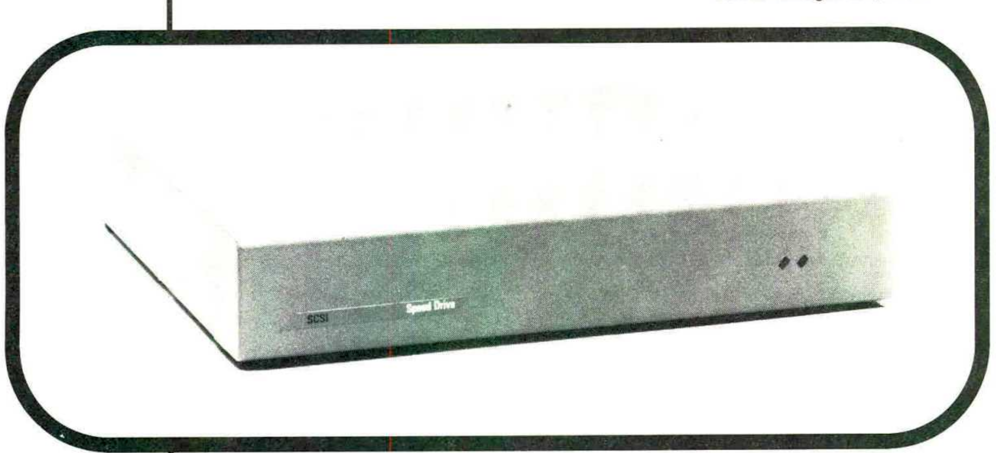
EASY ADRESS 3.0 [1]
SD 16 DM 15,-

- INFO -

Sonderdisks sind, obwohl der Preis auf gegenteiliges schließen läßt, keine Public Domain Programme. Sonderdisketten haben ein Copyright. Im Preis ist eine Beteiligung der Programmautoren enthalten. Sonderdisks ermöglichen den Usern, qualitativ hochwertige Software zu einem kostengünstigen Preis zu erhalten, wie die bisherigen Sonderdisks beweisen. Sonderdisks beinhalten Programme aus den verschiedensten Bereichen (z.B. Utilities, Grafik, Schulung, Spiele, DFÜ) - hier gibt es keine Einschränkung. Vielleicht haben auch Sie ein Programm für die Serie Schreiben Sie uns.

MAXON Computer, Industriestr. 26
'Idee Sonderdisk', 6236 Eschborn

SCSI Speed Drive Festplatten



siehe Testbericht ST Computer 4/90

- *SCSI Speed Drive Festplatten — eine der schnellsten und leisesten Festplatten für den Atari ST. 1 Jahr Garantie, 7 Tage Rückgaberecht, 49 MB 28 ms und 85 MB 24 ms.*
- *Ultra Speed Drive 42 MB, 19 ms, 64 KB Cache, Ultra Speed Drive 80 MB, 19 ms, 64 KB Cache — 2 Jahre Garantie*
- *Neu: Supral Speed Drive 80 und 110. Festplattten, die an Leistungsfähigkeit nicht mehr zu übertreffen sind!*
- *Neu: Ab sofort SCSI Speed Drive Wechselplatten lieferbar.*
- *Neu: 155 MB SCSI Speed Drive Streamer, Übertragungsrate 6,5 MB/Minute*
- *Nicht nur Bestellungen werden zu 95 % innerhalb von 24 Stunden ausgeliefert, auch technische Überprüfungen, Anpassungen und Reparaturen brauchen selten länger. Wer sonst bietet das?*

Leistungsdaten: Die Verbindung eines reinen SCSI-Hochgeschwindigkeits-Hostadapters und die Verwendung von SCSI-Festplatten ermöglichen Geschwindigkeiten, die bisher noch nicht erreicht wurden. In der Praxis ergeben sich Geschwindigkeitssteigerungen zwischen 30 — 60%. Die Festplatte ist 100% kompatibel zu den original Atari ST Festplatten. Das heißt: Sie können auch andere Harddisktreiber oder den original Atari Harddisktreiber benutzen. PC Speed, PC Ditto, Aladin usw. sind auf unserer Festplatte selbstverständlich lauffähig. Desweiteren ist in der Festplatte eine Echtzeituhr integriert. Die Festplatte wird mit einer sehr umfangreichen Software ausgeliefert.

DMA-Port: Der DMA-Port der Festplatte ist herausgeführt und komplett gepuffert. Das macht den Anschluß weiterer DMA-Geräte (Atari Laserdrucker, weitere Festplatten etc.) möglich.

Die Technik: Durch eine besondere Art der Luftzirkulation wird die Festplatte ohne störenden Lüfter betrieben und die Laufgeräusche der Festplatte optimal unterdrückt. Das macht die Festplatte zu eine der leisesten Festplatten für den Atari ST.
Das Gehäuse entspricht in Design und Abmessung dem Mega ST. Durch die robuste Ausführung kann es auch als Monitoruntersatz verwendet werden. Das Netzteil (VDE, GS) verfügt über 65 W und kann auch eine zweite interne Festplatte versorgen. Alle Festplatten verfügen über einen AUTO Park und sind mit einer speziellen Pufferung ausgestattet, die vor Schäden der Festplatte schützen, die durch kleine Stöße entstehen können.

Die Software: „SCSI TOOLS" ist ein bisher einzigartiges Softwarepaket, das in Leistung, Zuverlässigkeit und Geschwindigkeit neue Maßstäbe setzt. SCSI TOOLS ist die erste HD-Software, die zum neuen Atari-Standard (AHDI 3.0) kompatibel ist und die neuen Möglichkeiten von TOS 1.4 nutzt. Hochgeschwindigkeitstreiber voll AHDI 3.0 kompatibel, beliebig große Partitionen, Sektorgröße veränderbar, variabler GEM DOS Cache Buffer, Turbo DOS Kompatiblitätsmodus, besonders ausgeklügelter Softwareschreibschutz, Booten von allen Partition per Tastendruck, zusätzliche Datensicherheit durch Sicherheitskopie der Verwaltungsinformationen, Ausmappen von defekten Sektoren auf Controler und GEM DOS Ebene, komfortable graphische Benutzerführung mit Help Funktion, mit TOS 1.6 (1040 STE) lauffähig, Speed Cache, Treibersoftware für integrierte Echtzeituhr, außergewöhnliches Back Up Programm.

7 Tage Rückgaberecht

Garantie, Service: Auf unsere Festplattensysteme gewähren wir 1 ganzes Jahr Garantie. Sagt Ihnen die Festplatte trotz unserer Qualität nicht zu, gewähren wir Ihnen ein siebentägiges Rückgaberecht unter Übernahme der Porto- und Verpackungskosten Ihrerseits.

Preise: Speed Drive 49 MB 28 ms 1498,- DM, 85 MB 24 ms 1798,- DM; Ultra Speed Drive 42 MB 19 ms 64 KB C 1498,- DM; 155 MB SCSI Streamer 2298,- DM

Hard & Soft A. Herberg

Bahnhofstr. 289 · 4620 Castrop-Rauxel · ☏ (0 23 05) 1 57 64 · Fax 1 20 22

In der nächsten ST-Computer lesen Sie unter anderem

Simula

Smalltalk und C++ sind eingefleischten Computerfreaks sicherlich kein unbeschriebenes Blatt. Beide Begriffe verbindet die objektorientierte Programmierung, kurz OOP. Aus Dortmund, der deutschen Hochburg der objektorientierten Programmierung, kommt Simula, ein weiteres OOP-Produkt, das wir Ihnen in der nächsten Ausgabe vorstellen wollen.

Technobox Drafter

Als vor einigen Jahren das CAD-Programm Campus auf dem Markt kam, sorgte es für einige Aufegung auf dem ST-Markt. Mittlerweile hat das Programm schon einige Metamorphosen durchgemacht. Der Hersteller Technobox hat sich jetzt entschlossen eine Small-Version namens Drafter herauszubringen, die sich aber dennoch ohne weiteres sehen lassen kann. Was Drafter leistet, berichten wir in der nächsten Ausgabe der Computer.

BS-Handel in neuer Version

Bei Bavaria-Soft aus dem weißblauen Süden handelt es sich ebenfalls um eine Firma, die schon seit der ersten Stunde des ST mit Ihrer Business-Software kräftig mitmischt. BS-Handel ist jetzt in der Version 3.0 erhältlich, so daß wir nicht versäumen wollen, Ihnen die Features der neuen Version aufzuzeigen. Lassen Sie sich überraschen.

Die nächste ST Computer erscheint am Fr., den 29.6.90

Fragen an die Redaktion

Ein Magazin wie die ST-Computer zu erstellen, kostet sehr viel Zeit und Mühe. Da wir weiterhin vorhaben, die Qualität zu steigern, haben wir Redakteure ein großes Anliegen an Sie, liebe Leserinnen und Leser:

Bitte haben Sie Verständnis dafür, daß Fragen an die Redaktion nur **Donnerstags von 14⁰⁰-17⁰⁰ Uhr** unter der Rufnummer 06196/481814 telefonisch beantwortet werden können.

Natürlich können wir Ihnen **keine** speziellen Einkauftips geben. Wenden Sie sich in diesem Fall bitte an einen Fachhändler. Wir können nur Fragen zur ST Computer beantworten.

Vielen Dank für Ihr Verständnis

Impressum ST Computer

Chefredakteur: Uwe Bärtels (UB)
Stellvertreter: Harald Egel (HE)

Redaktion:
Uwe Bärtels (UB)
Harald Egel (HE)
Joachim Merz (JM)
Martin Pittelkow (MP)

Redaktionelle Mitarbeiter:
C.Borgmeier (CBO), Claus Brod (CB), Ingo Brümmer (IB), Derek dela Fuente (ddF), Stefan Höhn (SH), Raymund Hofmann (RH), Dieter Kühner (DK), Claus P. Lippert (CPL), Markus Nerding (MN), Chr. Schormann (CS), R.Tolksdorf (RT), Thomas Werner (TW)

Autoren dieser Ausgabe:
M.G. Berberich, D.Beyer, D.Brockhaus, J.Decker, A.Esser, R.Esser, H.G.Fischer, Dr.W. Kohlmeier, T.Luhm, J.Piater, G.Schabrun, M.Schneider, R.Stachs, M.Ullrich

Auslandskorrespondenz:
C.P.Lippert (Leitung), D.Dela Fuente (UK)

Redaktion: MAXON Computer GmbH
Postfach 59 69
Industriestr. 26
6236 Eschborn
Tel.: 0 61 96/48 18 14, FAX : 0 61 96/4 11 37

Verlag: Heim Fachverlag
Heidelberger Landstr. 194
6100 Darmstadt 13
Tel.: 0 61 51/5 60 57, FAX : 0 61 51/5 56 89 + 5 60 59

Verlagsleitung:
H.J.Heim

Anzeigenverkaufsleitung:
U.Heim

Anzeigenverkauf:
K.Margaritis

Anzeigenpreise:
nach Preisliste Nr.5, gültig ab 1.3.90
ISSN 0932-0385

Layout:
Martin Lowack,
Manfred Zimmermann

Titelgestaltung:
Gunter Wenzel

Fotografie:
Martin Lowack

Illustration:
Manfred Zimmermann, Martin Lowack

Produktion:
K.H.Hoffmann, B.Kissner

Druck:
Frotscher Druck GmbH

Lektorat:
V.Pfeiffer

Bezugsmöglichkeiten:
ATARI-Fachhandel, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag

ST Computer erscheint 11 x im Jahr
Einzelpreis: DM 8,-, ÖS 64,-, SFr 8,-
Jahresabonnement: DM 80,-
Europ. Ausland : DM 100,- Luftpost : DM 130,-
In den Preisen sind die gesetzliche MWSt. und die Zustellgebühren enthalten.


COMPUTER VIREN
Das neue
Anti Viren Kit 3.0
von G DATA
Jetzt auch für MS DOS*
und kompatible Computer:
ANTI VIREN KIT PC: DM 198,–**
Version für PC Speed: DM 99,–**
* MS DOS ist ein eingetragenes
Warenzeichen der Microsoft Corp
** Unverbindliche Preisempf.
Kostenloser
Update-Service
für registrierte Benutzer.
Mit Harddisk-Überwachung.
Diagnose von Link-Viren.
Das meistverkaufte
Anti-Viren-Programm
für ATARI ST Computer
Anti Viren Kit 3.0
Anti Viren Kit 3.0
DM 99,–
(unverbindl.
Preisempfehlung)
PROGRAMMABSTURZ!
DATENVERLUST!
Neue Version mit
Harddisk-Schreibüberwachung
und Virenscannung
Anti Viren Kit 3.0
bei Ihrem Atari-Händler
oder direkt bei
G DATA
Bundesrepublik:
Siemensstraße 16, D-4630 Bochum 1
Telefon. Bestellungen: 0 23 23/38 98 58
Versand zzgl. DM 5,– Portopauschale
per Nachnahme oder Vorausscheck.
Schweiz:
Computertrend, Langstr. 31, CH-8021 Zürich
Österreich:
CSR Computer Rittner, Hauptstraße 34,
A-7000 Eisenstadt